MUNDO ATARI

Número 3
AGOSTO 1987
$ 300

ST MIDI



PROGRAMA PARA PEQUES

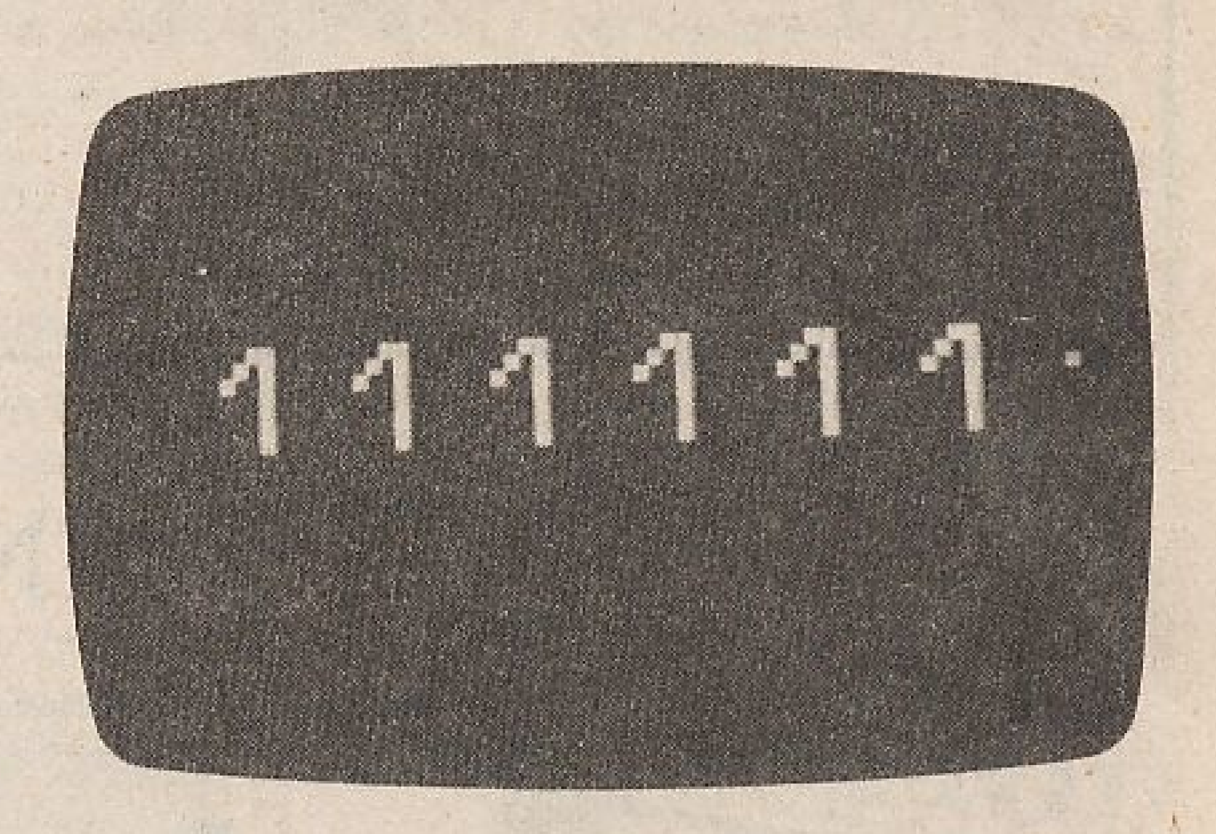
Y MUCHO MAS

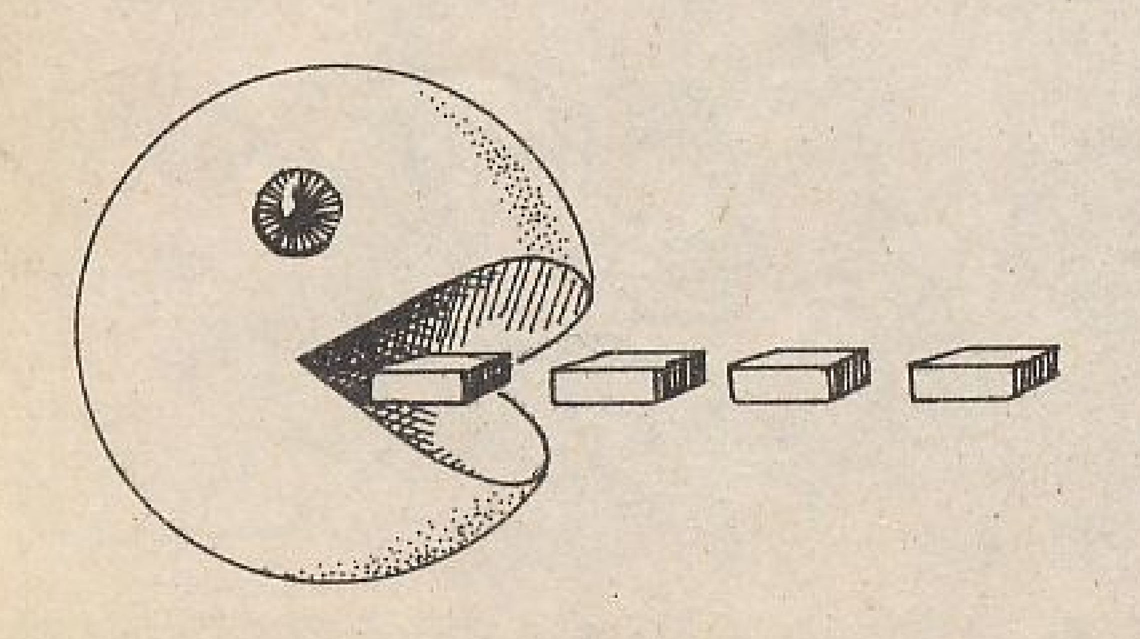
MUNDO ATARI®

PUBLICACION PARA USUARIOS DE MICROCOMPUTADORES ATARI







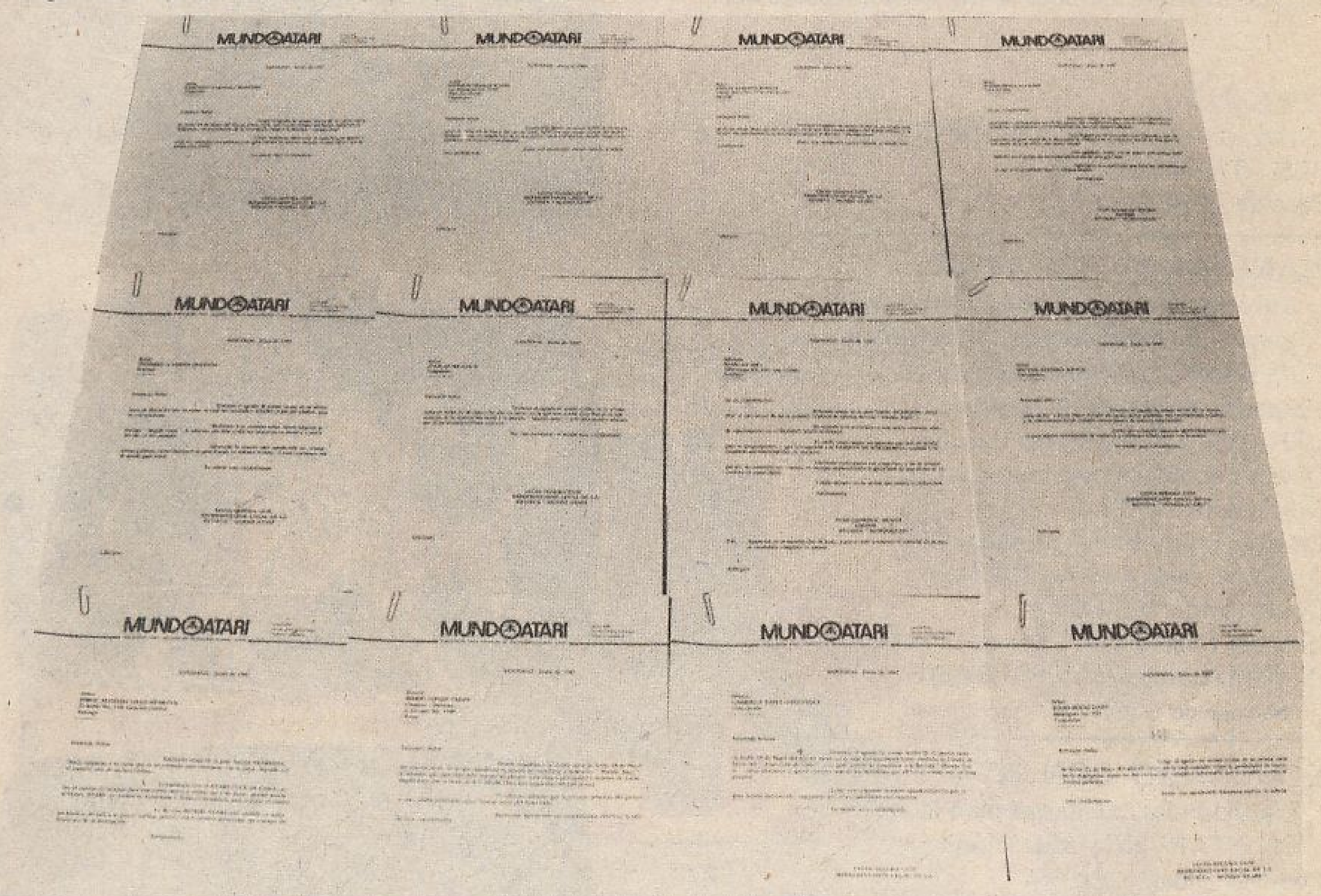




Amigos lectores:

La correspondencia recibida se responde individualmente. La cantidad y diversidad de ellas demora el proceso. Rogamos tener paciencia.

Sugerimos agregar en sus próximas cartas sus datos personales y otros para saber más de ustedes.



NUMEROS ATRASADOS



Editorial

ESTA editorial está dedicada a tí, amigo atariano, que día a día ocupas horas en conocer y aplicar esta fabulosa máquina de enseñanza. Conocemos tus desvelos y desencantos pasajeros, por la falta de información o los resultados inesperados. Nosotros, a su debido tiempo conocimos también esas experiencias.

Pero ustedes cuentan hoy con el apoyo de MUNDOATARI, publicación dedicada exclusivamente a cumplir estos objetivos.

Nuestra llegada a todo Chile,en esta primera etapa se encuentra cubierta por la distribución periódica nacional a través de los kioskos establecidos. Y la respuesta de nuestos lectores se manifiesta al examinar la procedencia de las muchas suscripciones.

El número de suscriptores actuales a MUNDOATARI es una muestra elocuente del interés y participación activa de los usuarios de computadores ATARI, hecho que nos permite proyectarnos con certeza en el futuro, cosa que en el pasado era una simple ilusión.

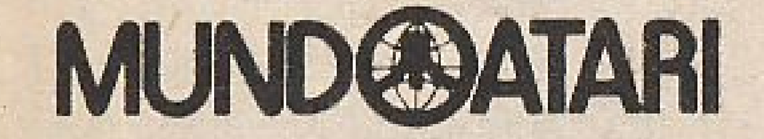
Todos ustedes ven en MUNDOATARI una posibilidad real de aprendizaje durante el mes, contenida en la información seria y motivadora de las numerosas columnas de continuidad.

En este número dejamos un testimonio de la potencia del computador ATARI y de la vigencia de la marca, al aplicar el manejo de las suscripciones con un antiguo ATARI 800, que cumple eficientemente el trabajo encomendado con un banco de datos que crece día a día.

Destacamos también en la columna PEQUES el programa "dibujando el uno", el cual abre una perspectiva diferente para el uso del computador ATARI a nivel de preescolares, para el desarrollo de la motricidad y del autoaprendizaje.

Finalmente, daremos una muestra de nuestra preocupación permanente por nuestros lectores con la presencia de MUNDOATARI en el stand de ATARI en SOFTEL'87, para recibirlos y atenderlos personalmente.

Hasta el próximo mes.



AGOSTO 1987 Precio: $ 300

Revista con información exclusiva para microcomputadores ATARI

Resol. Exenta No. 360/6-5-1987

Editor: Iván Gjurovic M.
Director: Adolfo Torrejón S.
Representante legal: Lucía Segura G.
Producción: SES Sistema
Casilla: 458, Correo Ñuñoa, Santiago
Teléfono: 2256579

Impresa por Editorial Antártica, quien sólo actúa como impresora.

Esta revista no mantiene relación de dependencia de ningún tipo con respecto a los fabricantes de microcomputadores ATARI ni sus representantes.

El contenido de la publicidad es responsabilidad de los avisadores.

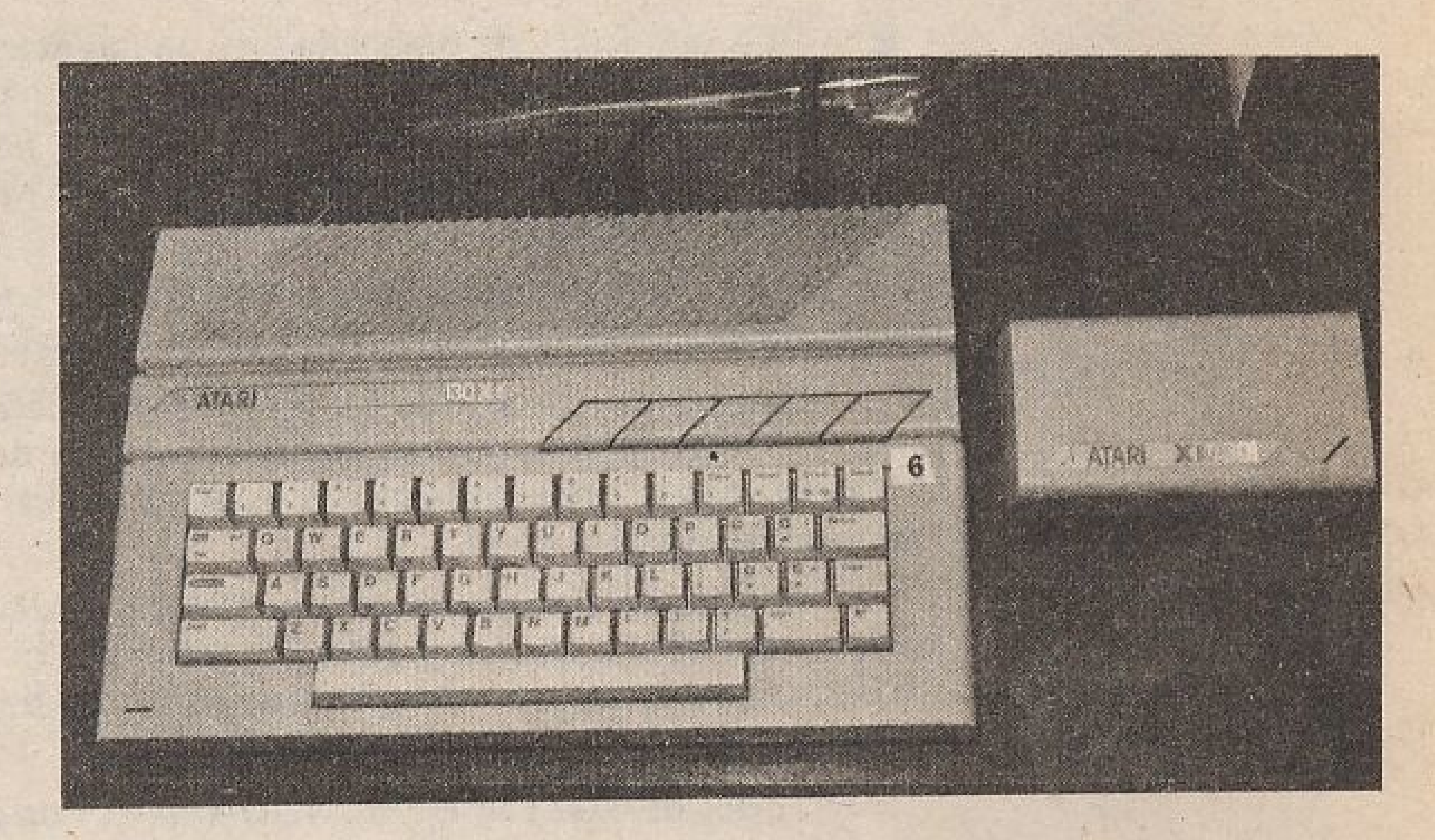
Modem telefónico ATARI XM-301

El MODEM (Modulador-Demodulador) XM-301 es un periférico ATARI, que le permite comunicar a su computador ATARI 800 XL ó 130 XE, a otro computador remoto, a través de las líneas telefónicas normales, a fin de que ambos computadores puedan "conversar", transmitiéndose información de un punto a otro.

La comunicación de computadores a distancia, lo llevará a usted y su equipo al fascinante mundo de las telecomunicaciociones, abriéndole un nuevo horizonte de conocimientos y aplicaciones a su computador ATARI.

Para establecer la comunicación entre computadores, además de necesitar el MODEM, se requiere disponer del software (programa) de comunicación apropiado para ello. En el caso del Modem XM-301, comercializado por COELSA COMPUTACION, éste incluye junto al equipo el programa en diskette X-TERM y uno en casete llamado AMODEM, que le permiten al Modem establecer la comunicación con otro computador —en cualquier parte del mundo— a través de una comunicación telefónica normal.

El XM-301 es un Modem eléctrico que se conecta en forma directa a su computador ATARI, a través del bus serial de la consola. Este dispone de la característica de autodiscado, lo que significa que desde el mismo teclado del computador usted puede discar el número telefónico al que desea llamar, logrando una operación más eficiente. Asimismo, el Modem contestará en forma automática si es llamado por otro usuario.



La velocidad de transferencia de datos desde un computador a otro se mide en una unidad denominada "baudios", correspondiente a número de bits por segundos (un byte es igual a 8 bits). Para el caso del XM-301, éste transmite a 300 baudios.

La adición de un Modem XM-301 a su configuración ATARI, abrirá un nuevo mundo lleno de sorpresas, aplicaciones y satisfacciones a su quehacer computacional, como ser, entablar comunicaciones con sus amigos, conectarse a un BBS (ver descripción del BBS ATARI COELSA) o a otros servicios que están apareciendo en el país.

Por último, a fin de no asustarse con el precio de este periférico, vale sólo algo más que una unidad de casete.

Telecomunicaciones

Comunicación. Según la Real Academia de la Lengua Española significa: ***"dar a conocer lo que uno sabe o conoce"***. Esto es fácilmente entendible dentro de los seres humanos, pero los computadores también tienen esa característica, la de poder comunicarse entre ellos.

La tecnología actual ha logrado que la comunicación entre computadores sea un acto extremadamente simple, tan simple como hablar con un amigo por teléfono.

Mucha gente se preguntará ¿para qué sirve hablar de computador a computador? La pregun-

ta es normal y obvia si es que uno no ha entrado al mundo de las comunicaciones. El objetivo de comunicar a computadores entre ellos es poder traspasar información, cualquiera que ésta sea, de un lado a otro y así poder estar en conocimiento de lo que está pasando en forma instantánea. Un buen ejemplo de esto es la información bursátil.

Para poder hacer "hablar" a su computador con otro, usted necesita un computador y un dispositivo llamado MODEM.

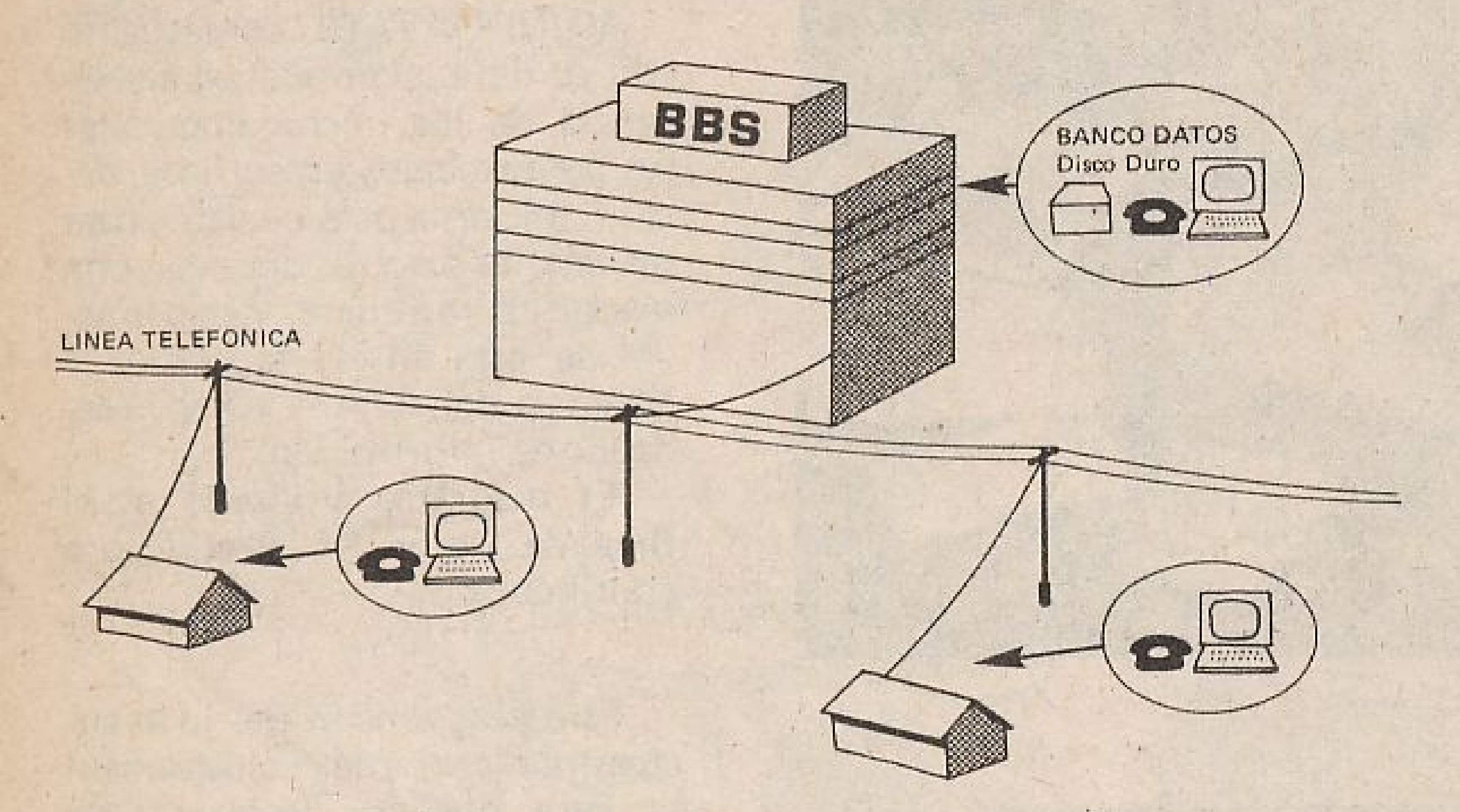
Este es un aparato que convierte las señales digitales de su computador en señales eléctricas que son enviadas a otro computador. Usted se estará preguntando cómo enviar dichas señales desde un punto a otro, pues es muy simple: por la línea telefónica de su casa.

El modem se conecta a la línea de teléfono y luego a su computador y ... voilá, usted se encuentra listo para efectuar una llamada desde su computador a otro, pero ¿adonde llamar?.

Usted puede llamar a un amigo que también tenga modem (previo aviso a su amigo), o bien llamar a algún BBS, donde podrá hacer muchas cosas interesantes.

Cuando usted llame a un amigo seguramente se pasarán algunos programas y posiblemente jueguen algún partido entre los dos. Si llama a un BBS entrará a un mundo de mensajes electrónicos, de cientos de programas de dominio público que usted podrá "llevarse" a su casa y luego usarlos. Podrá "enviar" programas que usted haya creado y desee dejarlos en el BBS para que otras personas lo usen, y muchísimas cosas más.

En el próximo número de esta revista encontrarán la primera parte de un artículo sobre comunicaciones, el cual les dará mayor información sobre comunicaciones vía computador.



BBS ATARI COELSA

BBS es una abreviación de Bilboard Bulletin System, lo que en español significa algo así como Sistema de Fichero Electrónico, el cual fue inventado y diseñado para que los usuarios de estos sistemas puedan efectuar intercambios de cualquier tipo de información.

El nuevo BBS ATARI COELSA es un sistema similar a lo anteriormente descrito que COELSA COMPUTACION ha puesto a disposición de los socios del ATARI CLUB de Chile, como otro servicio más a sus usuarios. El sistema le permite entre otras cosas:

– Enviar "mensajes electrónicos", y así poder conversar con otros usuarios sobre cualquier tema de interés común.
– Pertenecer a "grupos de interés" donde las personas conversan electrónicamente de sus mismos intereses.
– Obtener programas de dominio público que se encuentren en las distintas áreas del BBS.
– Dar a conocer a otros usuarios, datos o programas de dominio público que usted posea y desee incorporarlos al BBS de ATARI COELSA.
– Estar enterado de las últimas noticias, nacionales e internacionales del MUNDO ATARI.
– Mesa redonda con algún tema de interés para un grupo de participantes expertos en dicho tema.
– Y mucho más.

Este BBS tiene algunas restricciones. No se pueden dejar mensajes de tipo político, religioso u obseno. Son motivo de expulsión para aquellas personas que dejen este tipo de mensajes.

Para accesar desde su casa al BBS ATARI COELSA con un modem (por ejemplo el XM301), usted debe llamar al **5569132** en el siguiente horario:

– **de lunes a viernes de 19:00 a 8:00 horas.**
– **sábados, domingos y festivos todo el día.**

Una vez conectado, el sistema le solicitará su código de socio del ATARI CLUB de Chile y su nombre. Solamente si usted es miembro del Club, podrá hacer uso de este servicio gratuito que entrega COELSA COMPUTACION a sus usuarios.

Lo invitamos a participar de esta fascinante experiencia que le ofrecen las comunicaciones. ¡No se arrepentirá!. ●

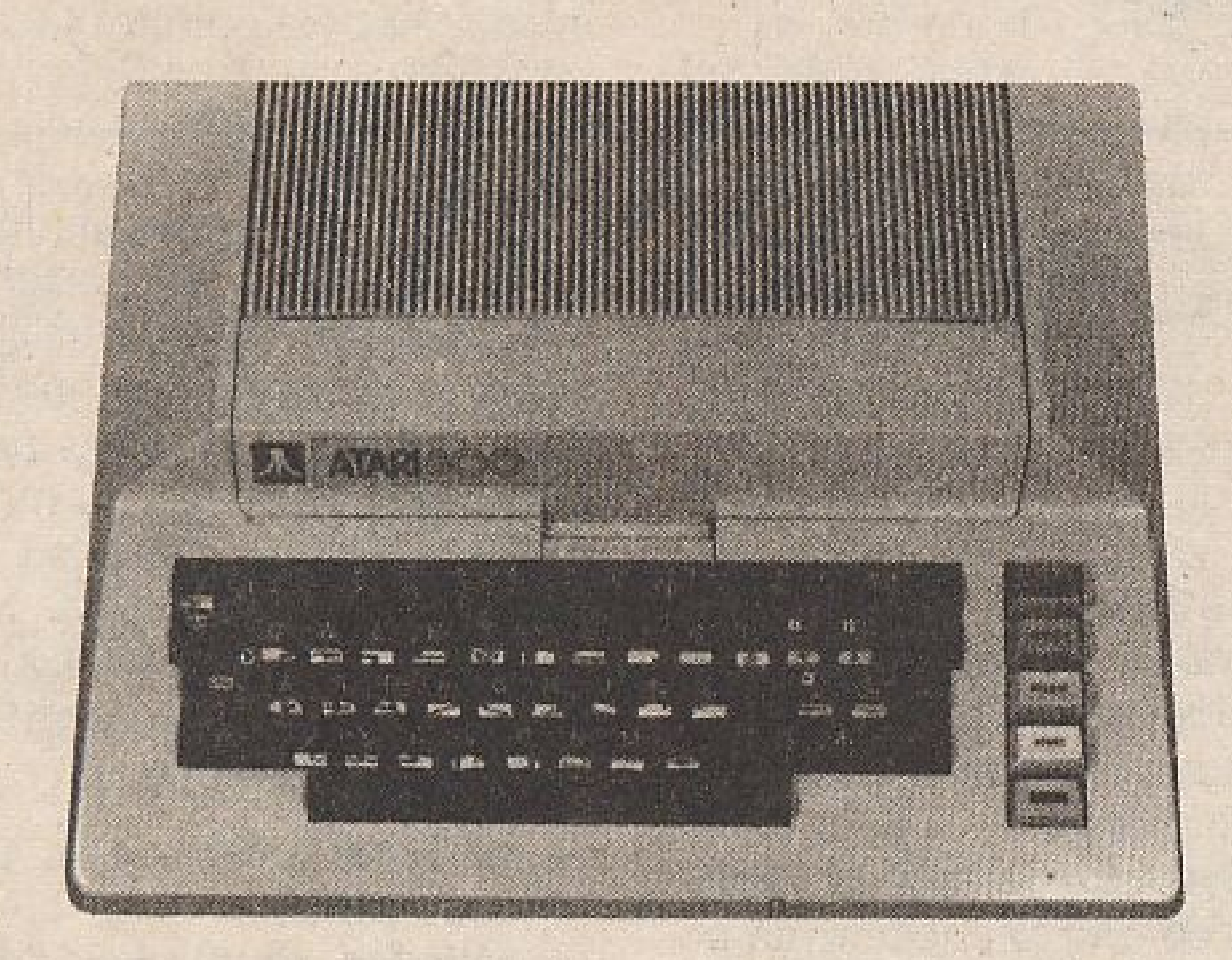
Base de datos XL

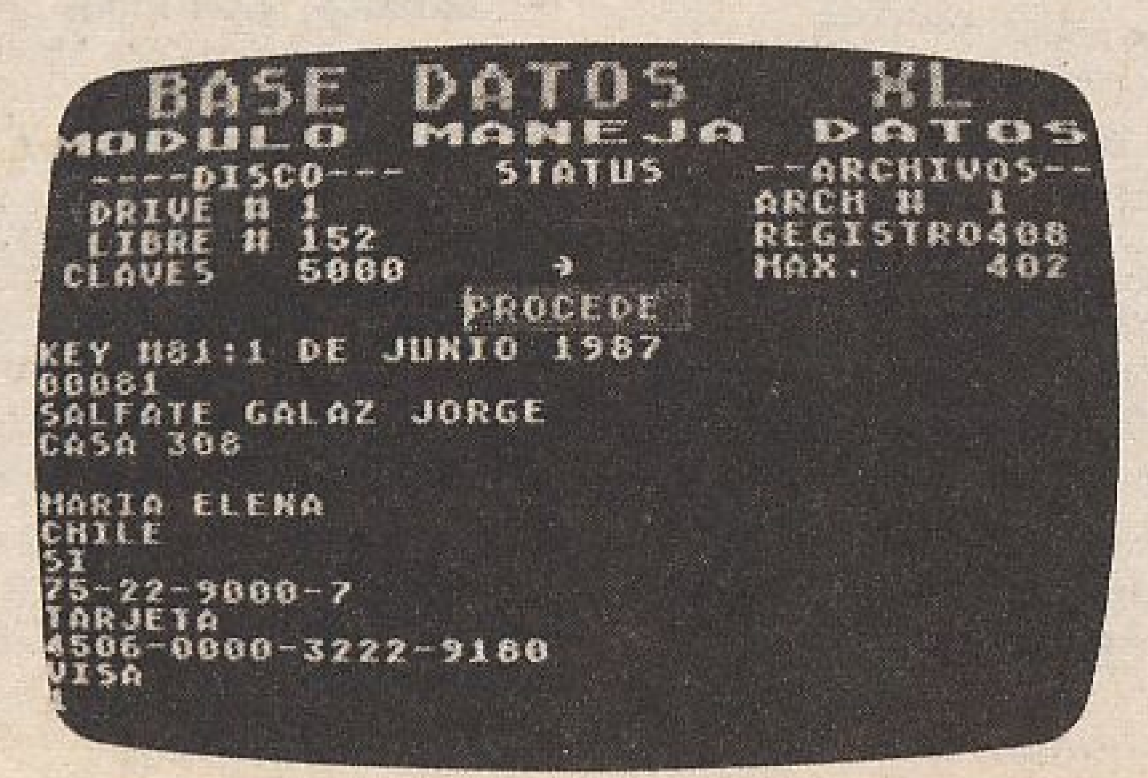


Amigos lectores del 800XL, ustedes probablemente no conocen el antiguo ATARI 800. La fotografía adjunta permite su presentación en plena labor.

MUNDOATARI, consecuente con su definición relativa a la vigencia de los microcomputadores para trabajos específicos, decidió iniciar la base de datos para manejo de suscripciones con esta magnífica máquina, demostrando de esta forma la validez y potencia del ATARI como ordenador de información.

El programa utilizado es el **Base de Datos XL.** Ver foto a continuación:





Este programa utilitario se encuentra disponible exclusivamente para nuestros lectores. Ver descripción en páginas de programas MUNDOATARI.

Este programa dispone en general de 5 programas modulares que se describen en el manual:

1. DATOS
2. INDICE
3. ARCHIVOS
4. IMPRESORA
5. UTILITARIO

Un ejemplo de su versatilidad es alguno de los informes elaborados por nosotros:

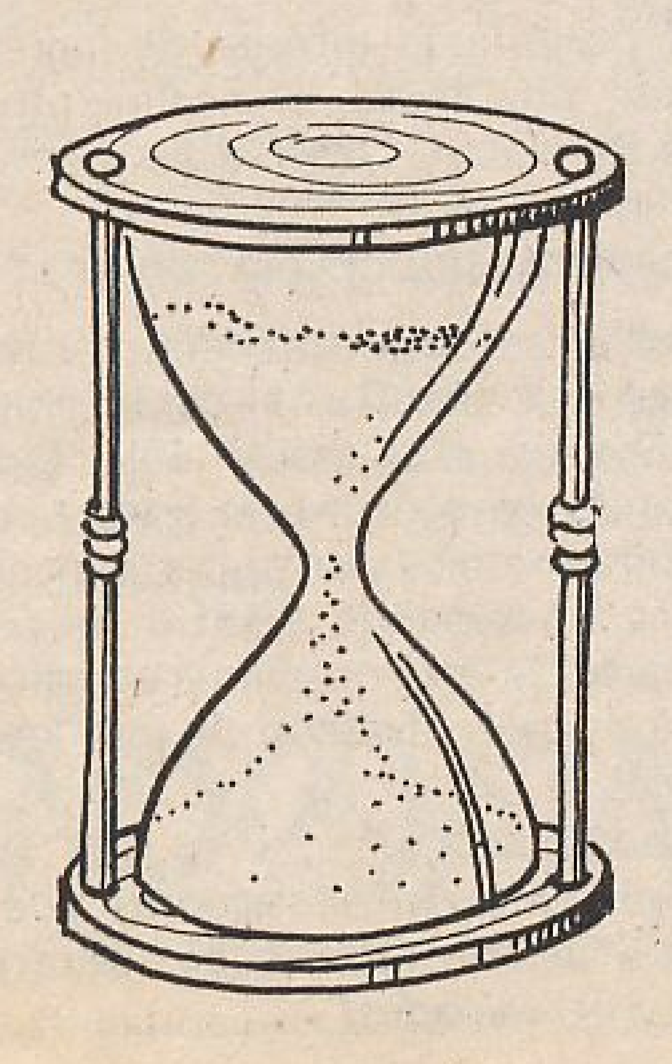
a) Informe de listado de suscriptores a MUNDOATARI como socios del ATARI CLUB, para aumentar el puntaje en su cartola para el concurso.
b) Informe de listado de tarjetas de crédito VISA para BANCARD.
c) Listado para despacho postal de revista a suscriptores.

Como consejo final, recomendamos mirar su ATARI desde este punto de vista y su productividad aumentará. ●

MANEJANDO TU ATARI

MUNDOATARI presenta a:

Prof. CHIP	Jefe de Proyecto
BENITO	Asistente
6502	Robot-junior



Guía BASIC

El computador ATARI es una fabulosa máquina para cálculos matemáticos. La columna GUIA BASIC comprueba este mes dicha capacidad con el programa SEGUNDOS.

El objetivo de este programa es calcular el número de segundos de vida de una persona.

Los datos de entrada son los siguientes:

- La fecha actual (día, mes, año)
- La fecha de nacimiento (ídem)

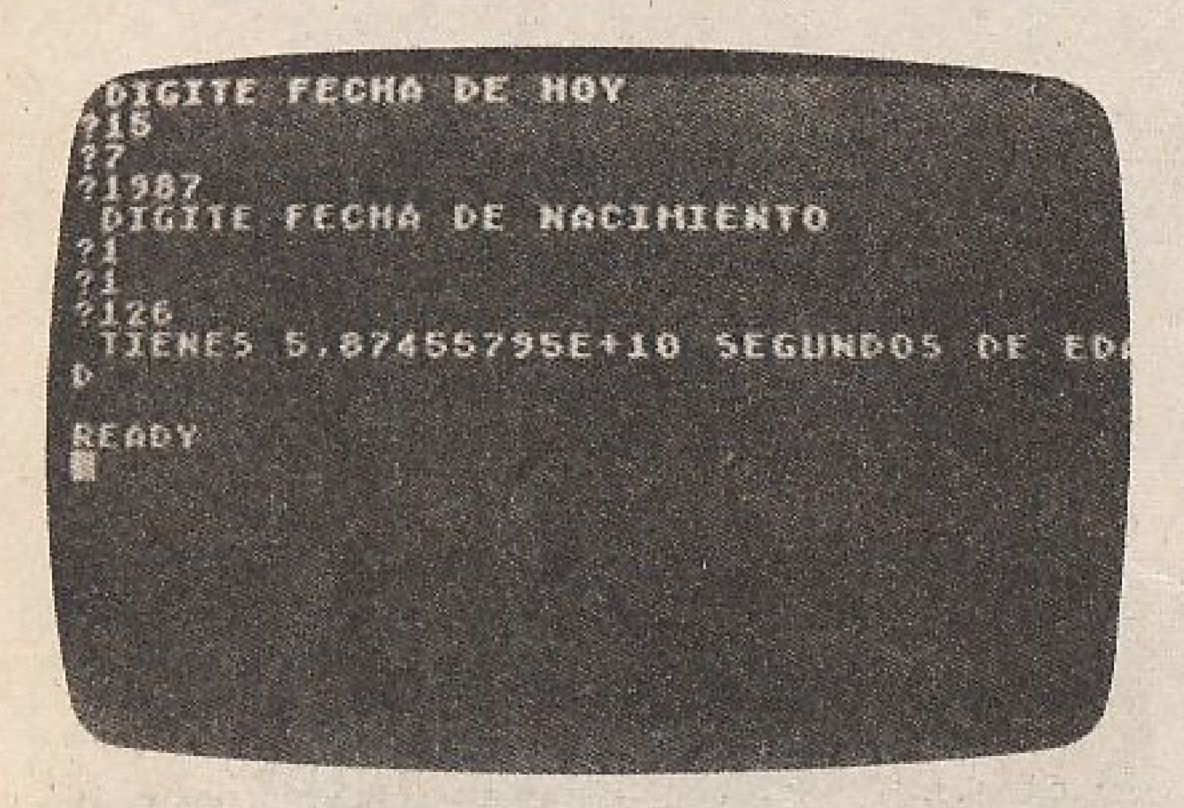
El ATARI procesa en segundos estos datos según las líneas de programa y . . .

. . . La salida es **el número de segundos.**

Ahora vamos a la acción. Digite cuidadosamente el listado siguiente:

```
10 REM SEGUNDOS GUIA BASIC AGOSTO 87
20 GRAPHICS 0
30 DIM N(12)
40 FOR K=1 TO 12:READ D:N(K)=D:NEXT K
45 DATA 31,31,28,31,30,31,30,31,31,30,
31,30
50 ? " DIGITE FECHA DE HOY"
60 INPUT D1,M1,A1
70 ? " DIGITE FECHA DE NACIMIENTO"
80 INPUT D2,M2,A2
90 D=D1-D2
100 M=M1-M2
110 A=A1-A2
120 IF D>=0 THEN 140
130 D=D+N(M1):M=M-1
140 IF M>=0 THEN 160
150 M=M+12:A=A-1
160 IF A/4<>INT(A/4) THEN 180
170 IF M1=3 AND M2=2 THEN D=D+1
180 D=365.25*A+30.24*M+D
190 S=D*24*60*60
200 ? " TIENES ";S;" SEGUNDOS DE EDAD
"
210 END
```

Ejecute el programa. Tenga cuidado de ingresar sólo números. La fotografía de pantalla muestra un formato de ingreso y salida para un ejemplo.



La edad ... desde otro punto de vista

Descripción de líneas de programa:

20 Inicializa modo gráfico cero
30 Dimensiona arreglo N para 13 subíndices (0 al 12)
40 Ingresa valores de días de meses al arreglo N
45 Datos de días de meses con desfase para arreglo
50-60 Mensaje de aviso de ingreso e ingreso múltiple de números en variables D1, M1, A1
70-80 Mensaje de aviso e ingreso de fecha de nacimiento en variables D2, M2, A2
90 Asigna a variable D la diferencia de días (D1 y D2)
100 Asigna a variable M la diferencia de meses M1 y M2)
110 Asigna a variable A la diferencia de número de años
120-130 Analiza el valor negativo de variable DIA. Suma el valor de días del mes anterior a fecha actual. Estudie en detalle este algoritmo
140-150 Analiza el valor negativo de variable MES. Suma el número de meses del año y disminuye en uno el número de años
160-170 Examina el año bisiesto
180-190 Relaciones de transformación a segundos de los años, meses y días
200 Línea para editar mensaje de salida.

Examine el valor de la línea de salida del valor del número de segundos (ver foto o pantalla), con toda seguridad la forma del número es en formato de punto flotante, pues es mayor que 10 elevado a 10.

Esta forma de escritura es práctica, pero para el programador inicial resulta incómoda. Por lo tanto el desafío de este mes es:

— Cambiar líneas para que el resultado se exprese en millones de segundos.

Otro punto de optimización del programa se refiere a la situación siguiente:

— En el momento de ingresar la fecha digite una letra y presione RETURN, el ATARI le envía un mensaje de error y el programa se detiene. Elimine esta posibilidad con las instrucciones pertinentes.

— Otra posibilidad errónea de ingreso de datos: Ingrese una fecha de nacimiento posterior a la fecha actual. Observe los valores que envía la salida de datos.
Elimine esta posibilidad con otras líneas.

Todas estas optimizaciones del programa forman parte del desafío 1 de la columna GUIA BASIC. Escriba a la casilla 458, Correo Ñuñoa, mencionando en el sobre el nombre de la columna y recibirá premios por su participación.

Ahora un segundo desafío de proyección es utilizar parte de este programa en calcular los intereses para el pago atrasado de una letra con vencimiento fijo.

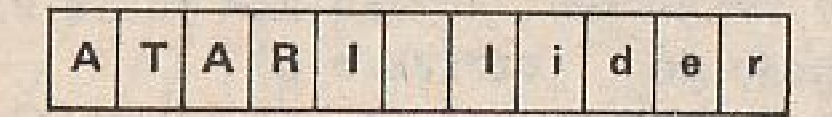
Amigo principiante, recuerde que esta columna es para usted. ¡Escríbanos para saber de sus intereses y necesidades!



Esta columna permanente está dirigida a todos los lectores para que formen una librería con rutinas, trucos y secretos. Se aceptarán colaboraciones para promover el intercambio de experiencias. Todas las rutinas que sean publicadas recibirán un premio de incentivo consistente en una Tarjeta de Referencia ATARI que contiene todos los datos de interés que un programador debe tener a mano.

La columna **Torpedo** de este mes proporcionará el inicio de una serie continuada de rutinas para la manipulación de strings.

Un **string** consiste en una cadena de caracteres alfanuméricos que se almacenan en una variable alfanumérica o en localizaciones temporales.

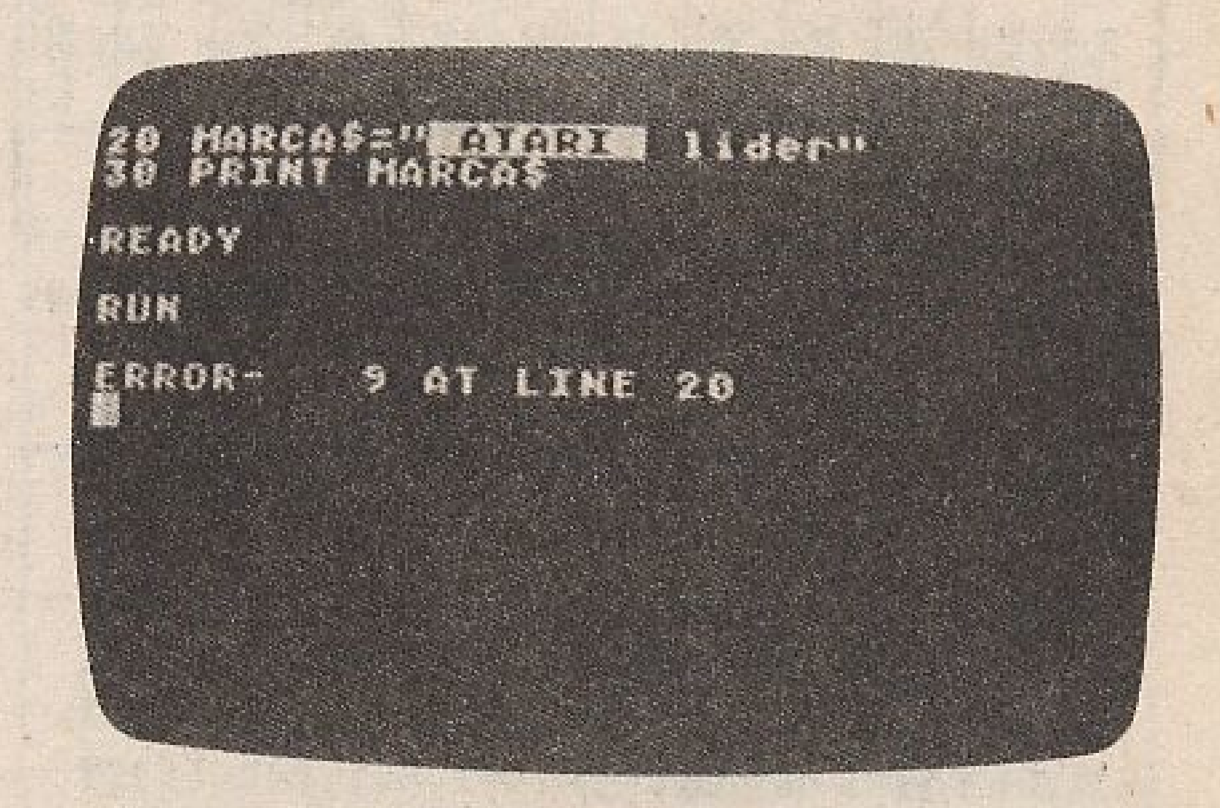


Digite en modo directo:

```
20 MARCA$=" ATARI  lider"
30 PRINT MARCA$
```

Ejecute con RUN y presione RETURN.

El ATARI BASIC interpreta un error como lo indica la foto siguiente:

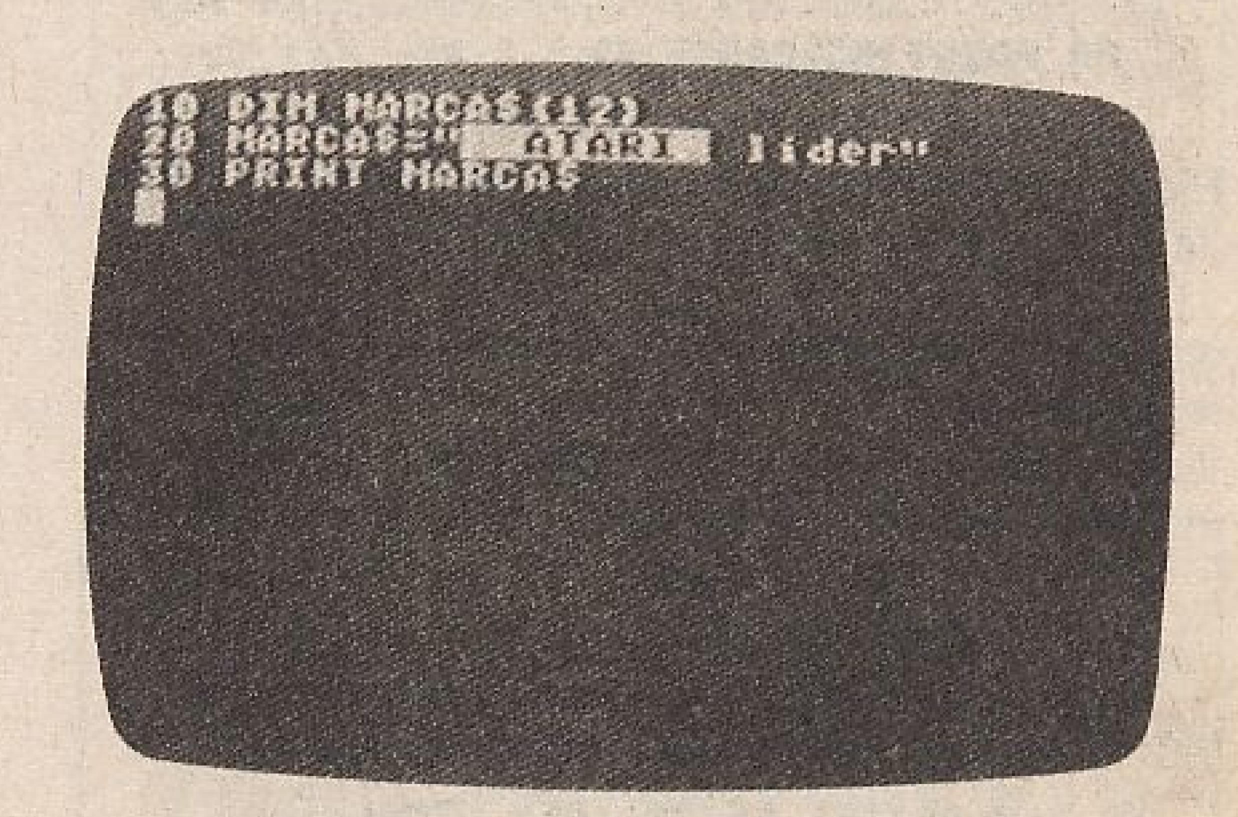


Para eliminarlo digite:

```
10 DIM MARCA$(12)
```

BORRE LA PANTALLA y digite LIST y después RETURN.

La pantalla aparecerá como lo ilustra la foto:



Aspecto del listado hasta la línea 30

Ejecute nuevamente el programa y comprobará la necesidad de dimensionar previamente las variables alfanuméricas.

Para este caso la dimensión previa fue de 12 localizaciones de memoria.

Para repasar conceptos de localizaciones de memoria, vea columna De Byte en Byte de MUNDOATARI No. 0.

Ahora se plantea una interrogante:

¿Cuáles son las localizaciones de memoria reservadas para la variable alfanumérica MARCA$?

Digite la línea:

```
40 PRINT ADR(MARCA$)
```

Al ejecutar el valor de retorno indicará la localización inicial de memoria reservada para la variable MARCA$.

LOCALIZACIONES DE MEMORIA

2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135
A	T	A	R	I		l	i	d	e	r	
1	2	3	4	5	6	7	8	9	10	11	12

Agregue a continuación la línea:

```
60 PRINT MARCA$(5)
```

Ejecute y observe en pantalla el segmento del string editado.

Este índice ubicado a continuación de la variable edita el contenido de las localizaciones a partir de la quinta.

Digite lo siguiente:

```
70 PRINT MARCA$(5,6)
```

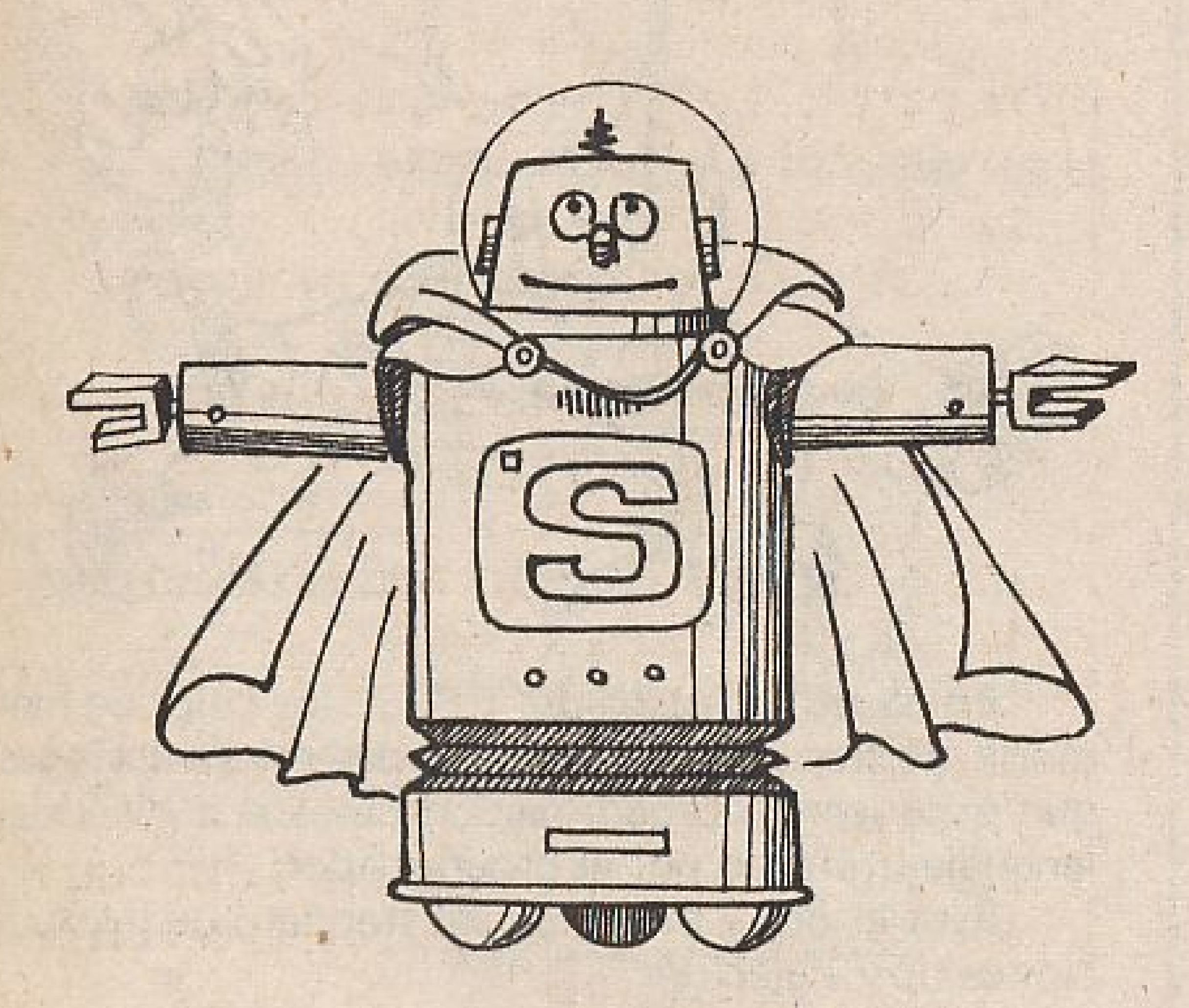
Ejecute y compare con el esquema. Edita en pantalla el contenido en memoria de posiciones 5 y 6.

Ahora sigue el programa final para este mes de la columna:

```
10 REM TORPEDO AGOSTO 87
20 REM Demostrativo para Subscript en
     un STRING posicion
30 DIM A$(10),B$(20):REM DIMensiona 2
     variables alfanumericas con 10 y 20
     caracteres respectivamente
35 ? CHR$(125):REM limpia la pantalla
40 A$="1234567890":REM asigna una cade
     na de caracteres a la variable A$
50 PRINT A$:REM edita el contenido de
     variable sin posiciones (TOTAL)
60 PRINT A$(5):REM edita el contenido
     desde posicion 5 hasta el final
70 PRINT A$(1,3):REM edita con doble
     posicion desde 1 hasta 3
80 ? :? :? :REM separador
99 REM Edita el contenido de acuerdo
     a Variable X
100 B$="ABCDEFGHIJKLMNOPQRST":REM Asig
     na a variable B$ 20 caracteres ma-
     yusculas ordenados
110 PRINT "B$    =";B$:REM edita el con
tenido de    variable B$
115 ? :REM Separador
120 X=1:REM Variable X para primera po
     sicion de STRING
130 FOR X=1 TO 9:? "B$(";X;")=";B$(X):
NEXT X:REM Edita el contenido definido
     por variable X
180 POSITION 0,22:? "PRESIONE una te
cla para continuar";
190 CLOSE #3:OPEN #3,4,0,"K:":GET #3,A
:REM pausa para analizar esta panta-
  lla
199 REM Edita el contenido de acuerdo
     a Variables X ,Y
200 Y=10:REM Variable Y para segunda
     posicion
205 ? CHR$(125):REM limpia pantalla
210 PRINT "B$     =";B$:REM edita el
     contenido de    variable B$
215 ? :REM Separador
220 FOR X=1 TO 5:Y=Y-1:REM Define limi
     te LOOP para variable X ,disminuye
     en 1 a variable Y
230 ? "B$(";X;",";Y;")=";B$(X,Y):NEXT
X:REM Edita el contenido definido por
     variable X
250 POSITION 0,22:? "PRESIONE una te
cla para continuar";
260 CLOSE #3:OPEN #3,4,0,"K:":GET #3,A
:REM pausa para analizar esta panta-
  lla
```

Para el próximo mes habrá una aplicación relacionada con el manejo de string con formato diferente de pantalla. ●

USR



La columna USR de este mes proporciona una utilísima rutina para ordenar elementos mediante el lenguaje de máquina.

```
5 REM PROGRAMA USR AGOSTO 87
6 REM Rutina para ordenar string a al-
ta velocidad
7 REM Formato de llamada es:
8 REM USR(ADR(SORT$),L,A,B,C,D,E,F)
9 REM L=longitud registro : A =loc ini
cial : B =loc. final
10 DIM SORT$(169),BUFFER$(256)
20 FOR I=1 TO 169:READ A:SORT$(I,I)=CH
R$(A):NEXT I
30 NR=100:RECLEN=10:DIM T$(RECLEN),MAS
TER$(NR*10)
40 PRINT "ЌCREANDO UN  STRINGS":POKE 7
52,1
50 FOR A=1 TO NR:FOR B=1 TO 10:T$(B,B)
=CHR$(65+RND(1)*25):NEXT B:PRINT A;"↑"
60 MASTER$((A-1)*10+1,A*10)=T$:NEXT A
70 PRINT "Ќ":FOR A=1 TO NR:PRINT MASTE
R$((A-1)*10+1,A*10):NEXT A
80 PRINT "Ќ↓PRESIONE UNA TECLA   ":GOS
UB 150:PRINT "↓ORDENADO"
90 L=RECLEN:A=ADR(MASTER$):B=A+NR*RECL
EN:C=B-RECLEN:D=ADR(BUFFER$):E=0:F=9
100 ADDR=41+ADR(SORT$):HBYTE=INT(ADDR/
256):LBYTE=ADDR-256*HBYTE:POKE 232,LBY
TE:POKE 233,HBYTE
110 SORT=USR(ADR(SORT$),L,A,B,C,D,E,F)
120 PRINT "↓LISTO":PRINT "↓PRESIONE UN
A TECLA":GOSUB 150
130 FOR A=1 TO NR:PRINT MASTER$((A-1)*
10+1,A*10):NEXT A
140 POKE 752,0:END
150 POKE 764,255
160 IF PEEK(764)=255 THEN 160
170 RETURN
180 DATA 104,104,104,133,240,104,133,2
42,133,244,104,133,241,133,243,104,133
,246,104,133,245,104,133,248,104,133
190 DATA 247,104,133,250,104,133,249,1
04,104,133,230,104,104,133,231
200 DATA 165,242,133,252,165,241,133,2
51
210 DATA 24,165,241,101,240,133,241,14
4,2,230,242
220 DATA 165,242,197,246,208,6,165,241
,197,245,240,29,164,230,177,241
230 DATA 209,251,240,13,176,223,165,24
2,133,252,165,241,133,251,24,144,212,2
08,196,231,240,207,24,144,229
240 DATA 160,0,177,251,145,249,200,196
,240,208,247,160,0,177,243,145,251,200
,196,240,208,247,160,0,177,249
250 DATA 145,243,200,196,240,208,247
260 DATA 24,165,243,101,240,133,243,14
4,2,230,244
270 DATA 165,244,197,248,208,7,165,243
,197,247,208,1,96,165,244,133,242,165,
243,133,241,108,232,0
```

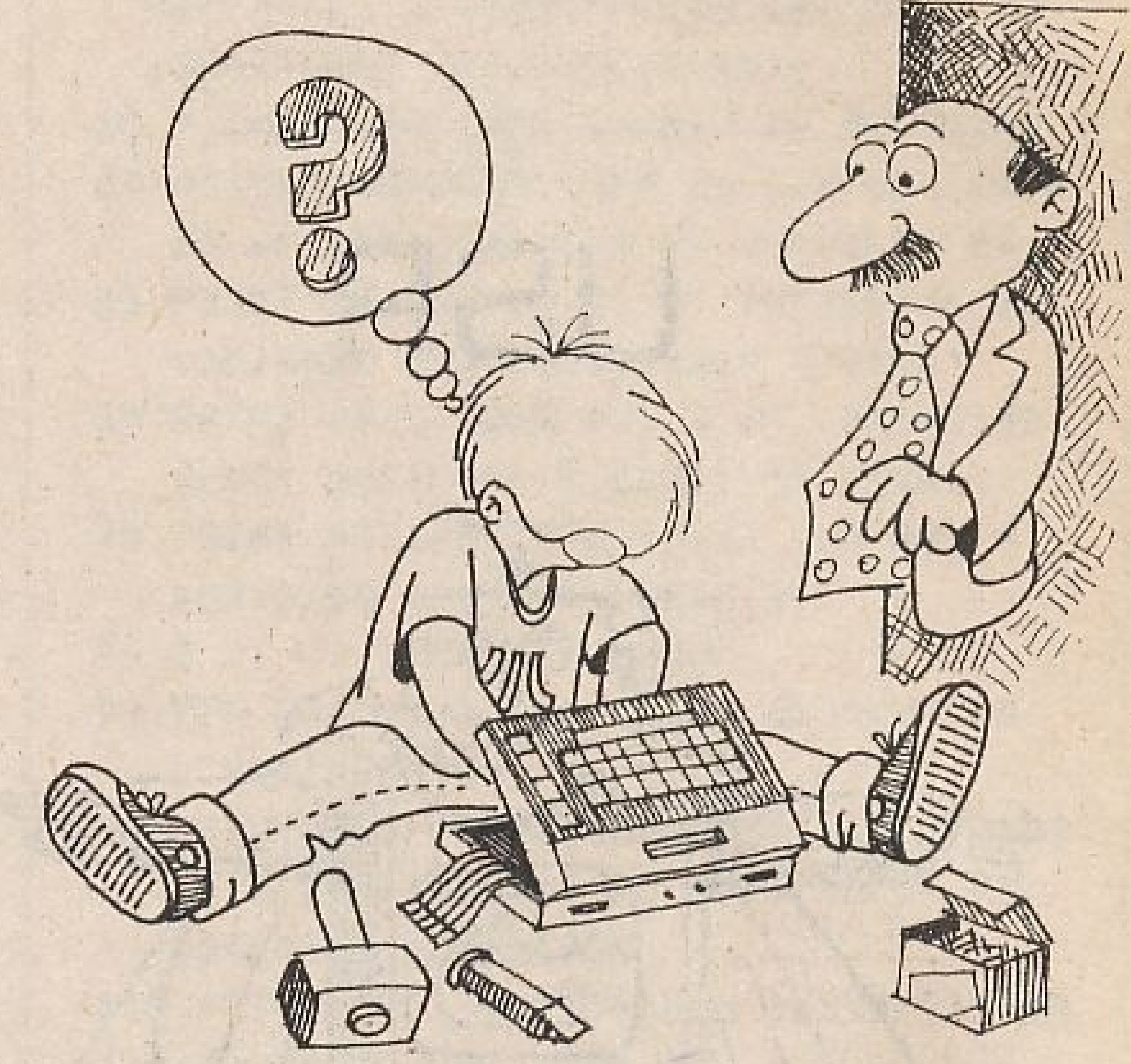
Ejecute el programa y disfrute la velocidad de la máquina.

Usted puede hacer uso de ella con esta rutina. Domínela y aplíquela en sus programas. Escríbanos contando sus optimizaciones.

Por último una tarea para nuestros **tuercas:**

— Escríbanos el significado de cada uno de los parámetros para la función USR de la línea 110 y el valor de retorno que se asigna a la función SORT. ●

Directo al 6502

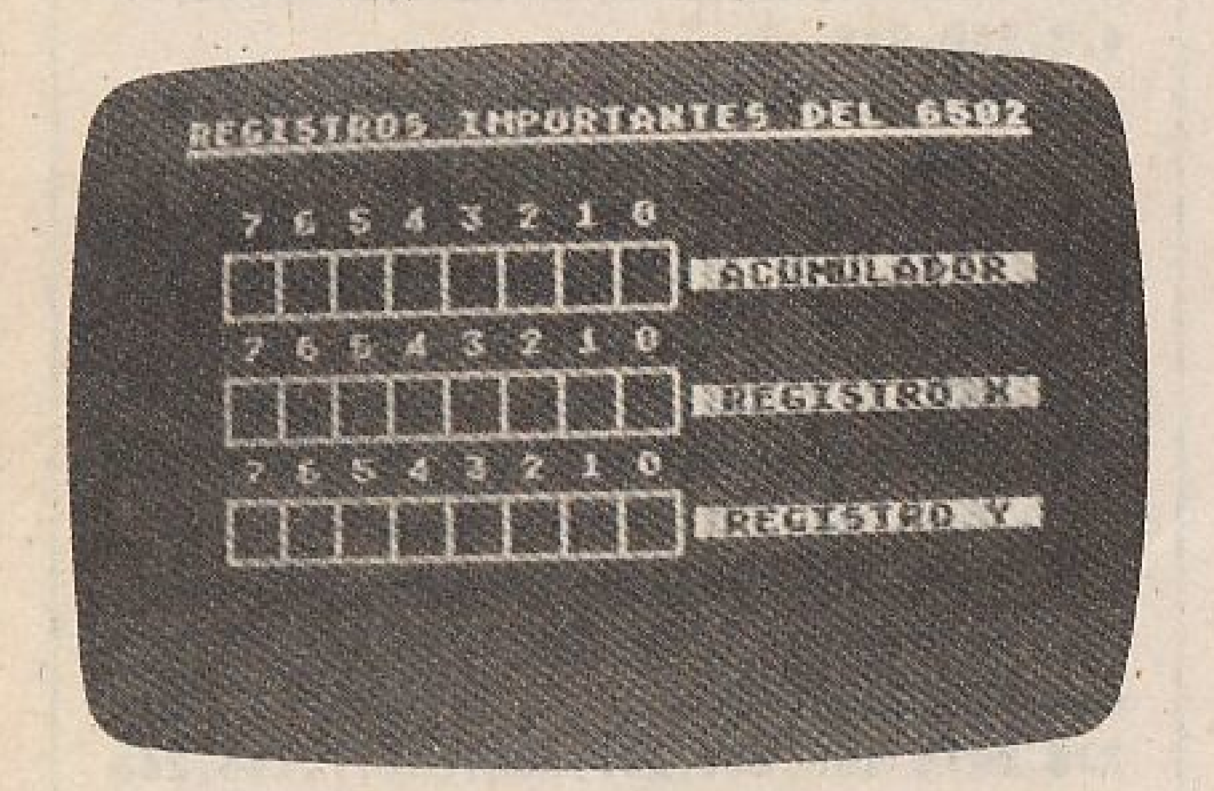


Los números iniciales de MUNDOATARI en la columna **Directo al 6502**, describen los diferentes sistemas numéricos que se resumen a continuación:

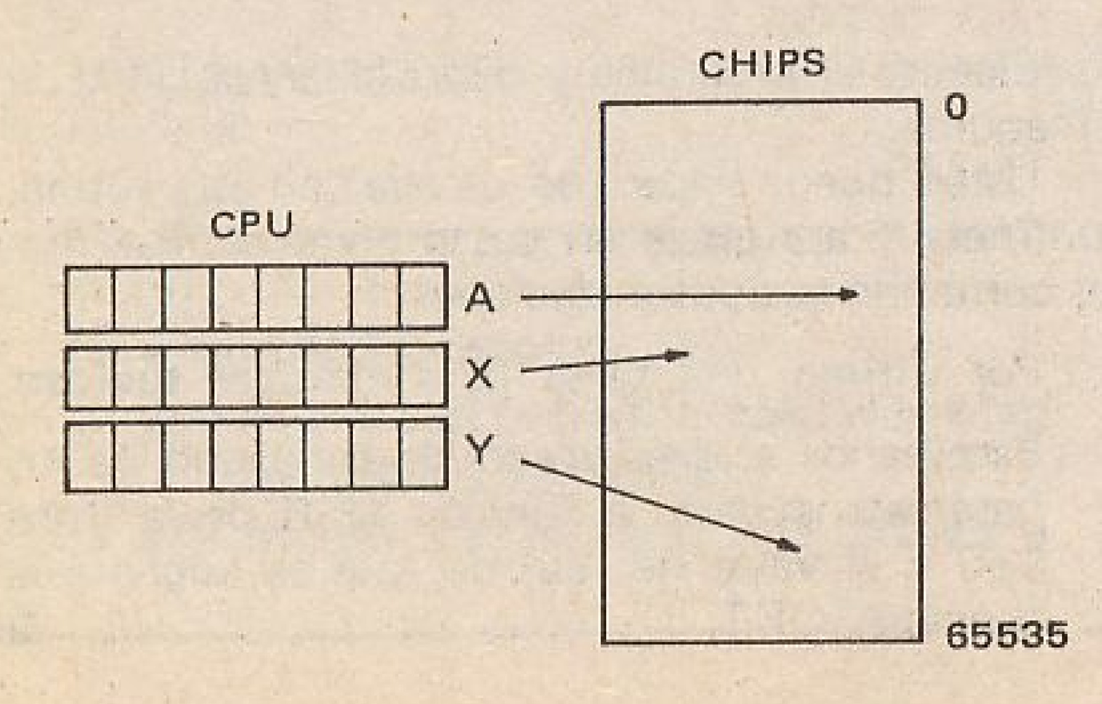
- **El sistema binario**, utilizado por la máquina.
- **El sistema decimal**, propio de los seres humanos.
- **El sistema hexadecimal**, puente de conexión entre los sistemas anteriores.

La estructura interna del ATARI contiene una Unidad Central de Proceso (CPU), que cuenta con tres registros principales:

1. **El acumulador**
2. **El registro X**
3. **El registro Y.** Todos ellos tienen 8 bits y por lo tanto pueden contener números enteros entre 0 y 255.



Por otra parte, en el interior, se encuentran los chips de memoria, que para el ATARI 800 son de 65536 bytes o localizaciones de memoria.



Entre los registros del CPU y los chips de memoria existen diversas operaciones e interacciones que producen la programación deseada a nivel del lenguaje utilizado por el programador.

Para el 6502 existe un set standard de instrucciones tipo a ejecutar.

Ejemplo: **Carga el acumulador.**

Usted al examinar un programa en ASSEMBLER nunca encontrará estas palabras, pero sí un símbolo equivalente: **LDA.**

¿Qué representa esta sigla?

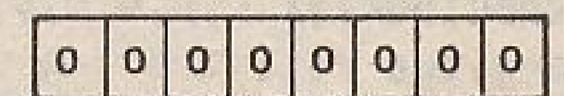
Es un código ayuda memoria denominado MNEMONIC.

En relación a la instrucción definida ¿qué valor puede cargar el acumulador?

La instrucción:

LDA #18

carga el acumulador con el valor 18 decimal. Es decir, el valor inicial cero que contenía el acumulador:



es reemplazado al ejecutar la instrucción anterior. El acumulador queda entonces con:



Este formato se define como **direccionamiento inmediato.**

ACTIVIDAD:

Complete el esquema del acumulador al ejecutarse las siguientes instrucciones:

ASSEMBLER

LDA # 35 ☐☐☐☐☐☐☐☐

LDA # 255 ☐☐☐☐☐☐☐☐

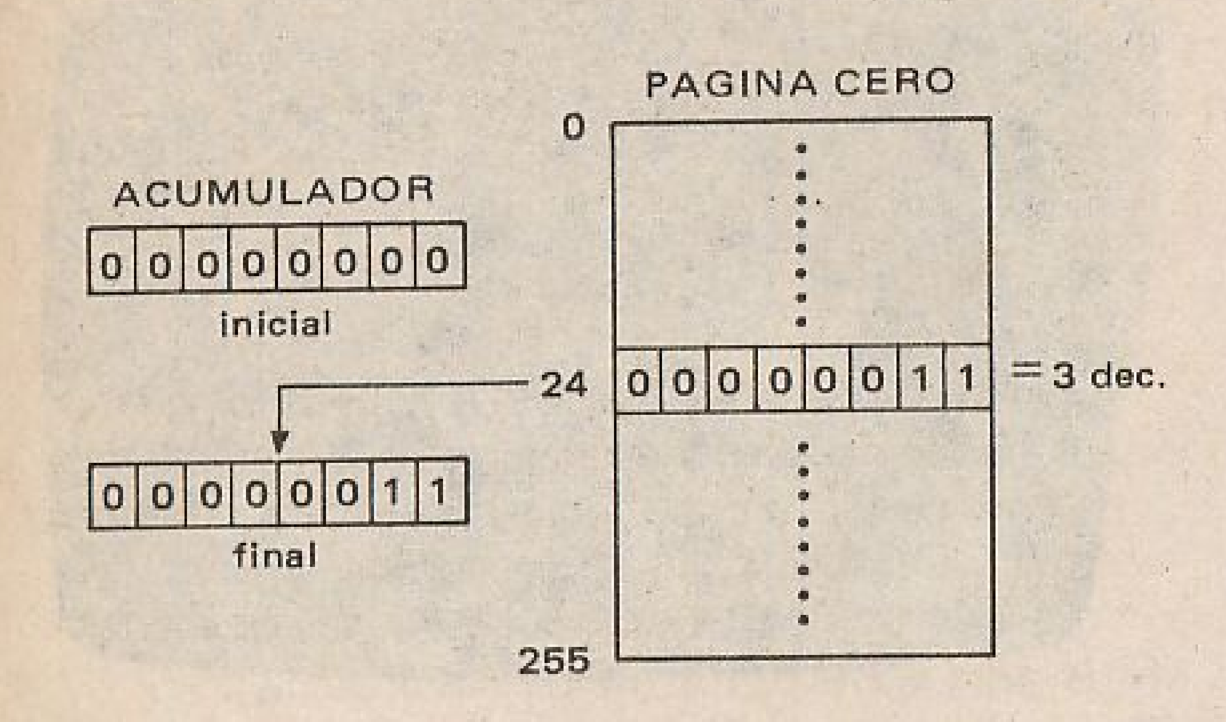
LDA #$ 18 ☐☐☐☐☐☐☐☐

El signo # determina el direccionamiento inmediato.

Examinemos otro formato para la instrucción LDA:

LDA $ 18

El esquema siguiente ilustra la ejecución de la instrucción:



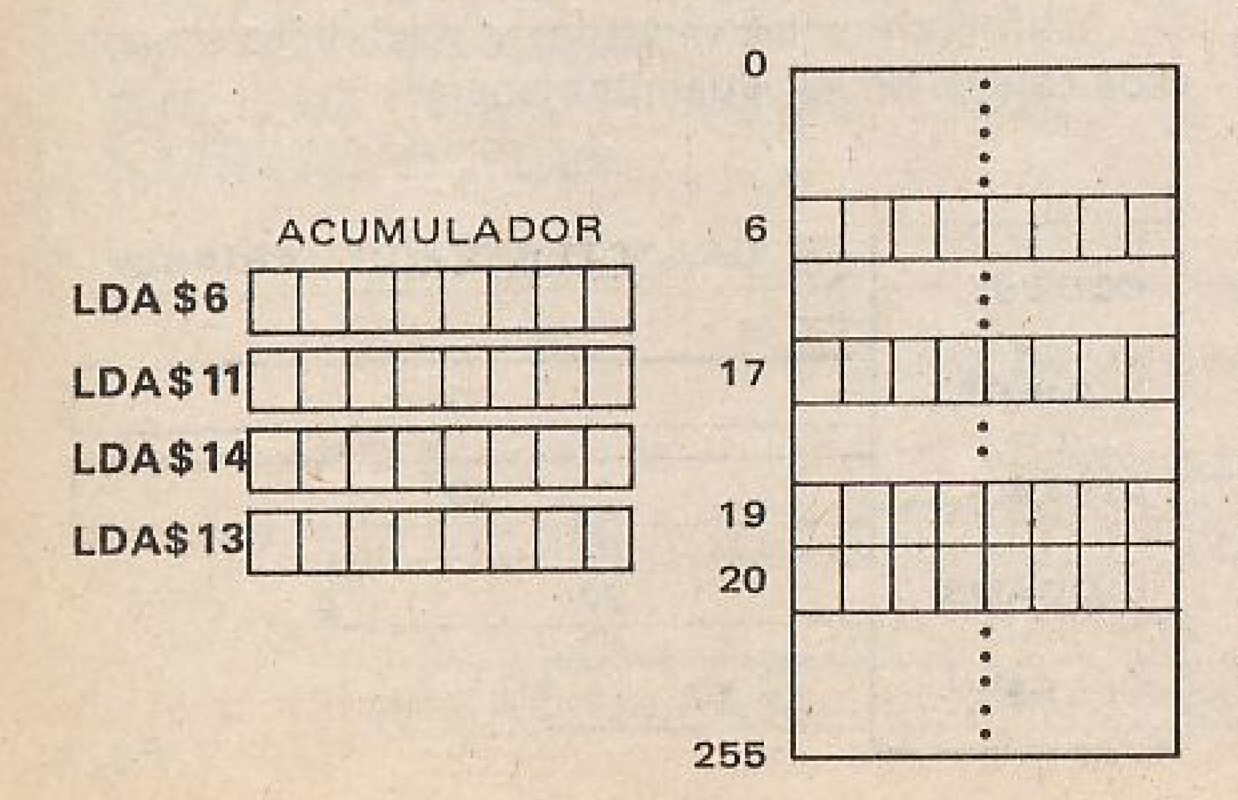
El valor presente en el acumulador al procesarse esta instrucción es el contenido actual de la localización 18 hexadecimal (24 decimal).

Este tipo de formato se refiere exclusivamente a la primera página de la memoria, que contiene punteros importantes para la observación del Sistema Operativo.

Este tipo de direccionamiento se conoce como **página cero,** por la referencia a dicha página de memoria.

ACTIVIDAD:

En el siguiente esquema determine el contenido del acumulador al procesarse las siguientes instrucciones:





PARTICIPANDO

Esta columna de contribuciones a MUNDO-ATARI publica el programa **Truco Secreto,** de Francisco Riquelme, socio de ATARI CLUB No. 65-24-7001-0 de Los Angeles.
Se trata de un interesante truco para escribir en pantalla con sonido y retardo.
La velocidad para editar en pantalla se controla con la pausa de línea 1040.

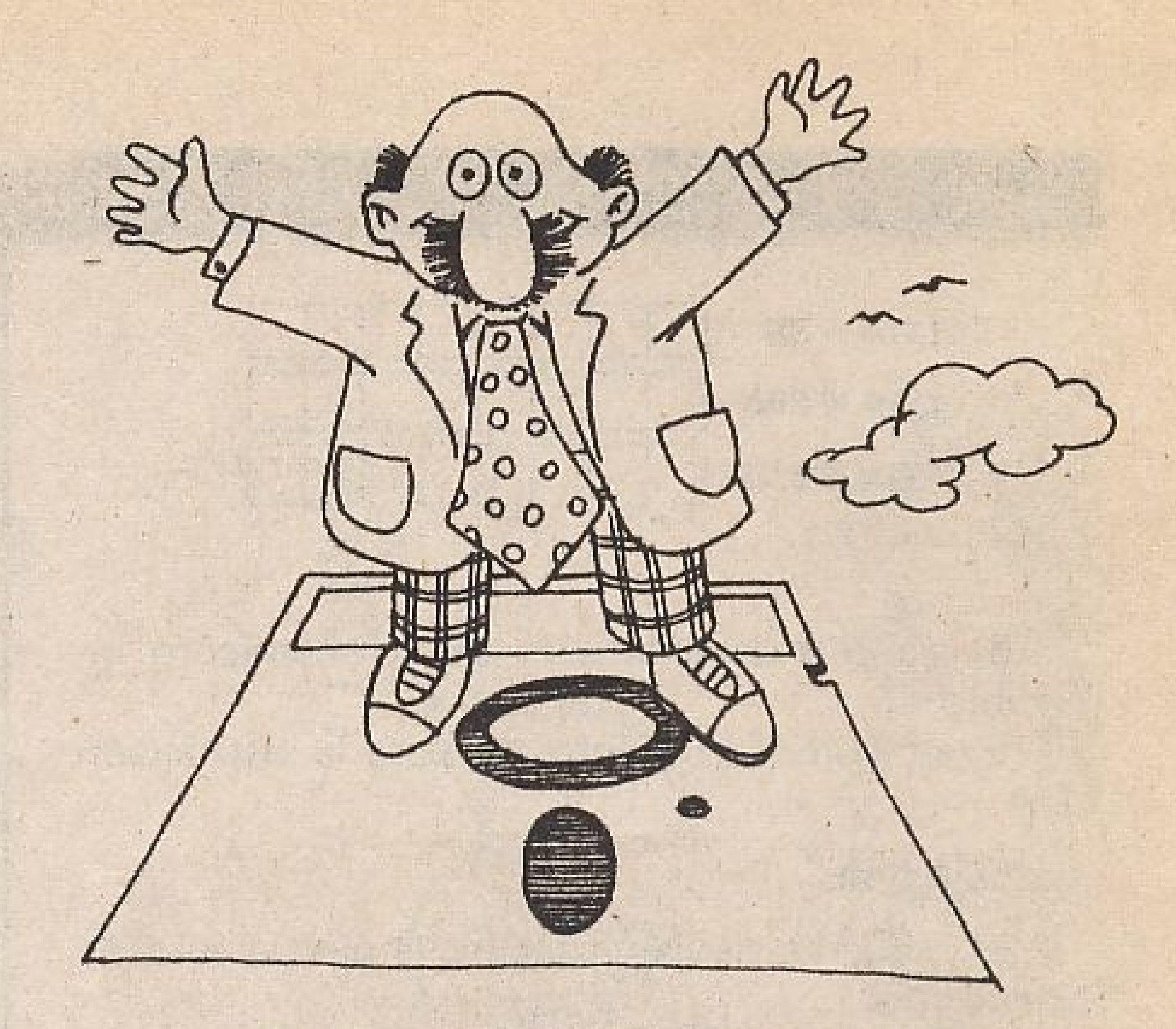
```
1 REM PROGRAMA TRUCO SECRETO
2 REM FRANCISCO RIQUELME
3 REM SOCIO # 65-24-7001-0
4 REM
5 REM LOS ANGELES
6 REM VII REGION
8 REM CHILE
9 REM *******************************
10 REM
20 DIM A$(20)
30 A$="HOLA MUNDO ATARI ":GOSUB 1000
90 END
100 FOR T=1 TO LEN(A$):? A$(T,T);:SOUN
D 0,23,30,15:SOUND 0,0,0,0:FOR A=1 TO
70:NEXT A:NEXT T
1000 FOR T=1 TO LEN(A$)
1010 PRINT A$(T,T);
1020 SOUND 0,23,30,15
1030 SOUND 0,0,0,0
1040 FOR PAUSA=1 TO 50:NEXT PAUSA
1050 NEXT T
1060 RETURN
```

NOTA:

Se solicita a los lectores que el envío de contribuciones se haga de acuerdo al formato de las líneas 1 a 10, y además envíen comentarios cortos al respecto.
A nuestro lector favorecido de hoy se le ha premiado con un programa en diskkette de BASIC Avanzado.

DISKETTE

Dominando la 1050



Amigos usuarios de la Unidad de Disco 1050, este mes resumiremos algunos conceptos de meses anteriores en relación a modos de operación con este periférico.

El modo 8 corresponde a un modo que permite escribir información en el diskette.

El modo 4 realiza el proceso inverso, es decir, lee información desde el diskette hacia la memoria.

Los IOCB son los blocks que controlan el traslado de la información y existen comandos del ATARI BASIC para su manejo, como son:

- OPEN
- CLOSE

El acceso al IOCB en una operación característica requiere de 4 parámetros:

Ejemplo:

OPEN #1, 8, 0, "D:MUNDO"

El primer parámetro que antecede al símbolo # identifica al IOCB que está ejecutando la operación.

El segundo parámetro identifica el tipo de operación y ya fue definido al inicio de esta columna.

El tercer parámetro no tiene un real sentido para nuestro periférico. Diferente es en el uso de la Unidad de Casete que define la velocidad de grabación.

El cuarto parámetro identifica al archivo del directorio con el que se establece la transferencia.

Con este preámbulo vamos a una situación problema:

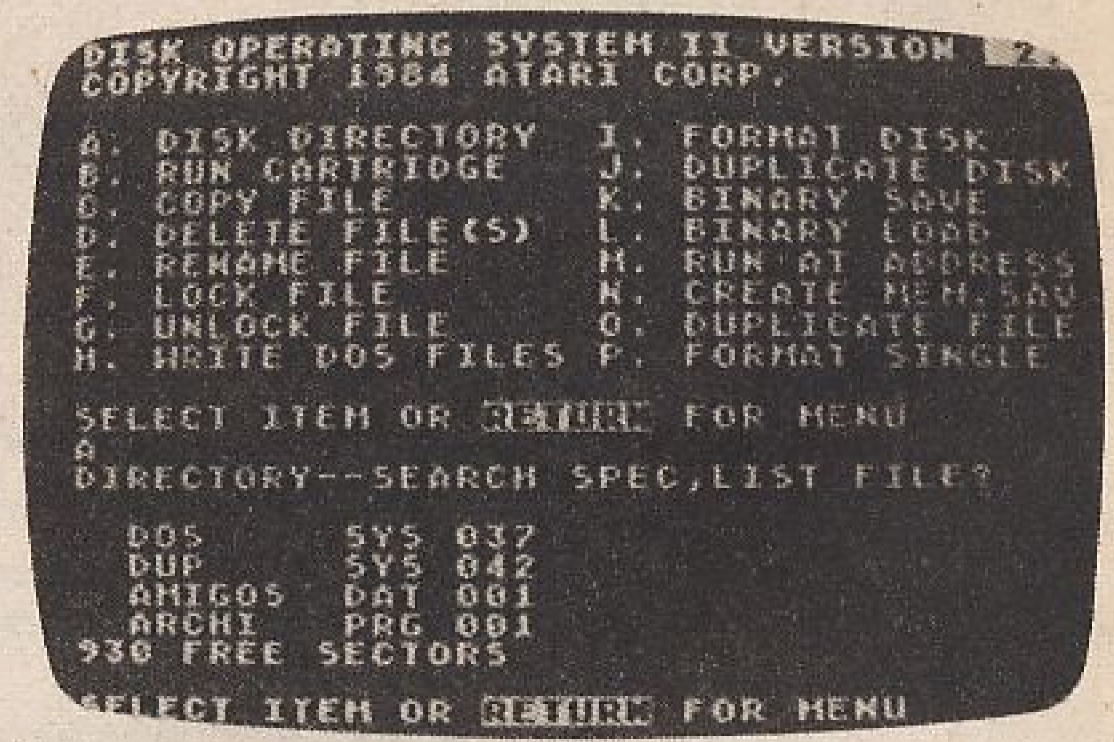
1. Formatee un nuevo diskette y agregue el DOS (ver números anteriores de MUNDOATARI).
2. Abra un archivo en el directorio de un diskette con el nombre AMIGOS.DAT

Digite la siguiente línea de programa:

```
1000 OPEN #1,8,0,"D:AMIGOS.DAT"
```

Salve esta línea con el nombre ARCHI.PRG (utilice instrucción LIST).

Examine el directorio del diskette y compruebe en el directorio. Este debe ser similar al de la fotografía siguiente:



Esto indica que en el directorio existe:

A. Un archivo con la línea de programa (1000) con el nombre ARCHI.PRG
B. Un archivo abierto para datos con el nombre AMIGOS.DAT

3. Cada registro del archivo lleva la información siguiente: (campos)

   a) Código
   b) Nombre
   c) Dirección
   d) Ciudad
   e) Teléfono

Para el ingreso de esta información en estos campos de cada registro se necesita de variables alfanuméricas.

Definiremos las variables y su dimensión para cada campo de la siguiente manera:

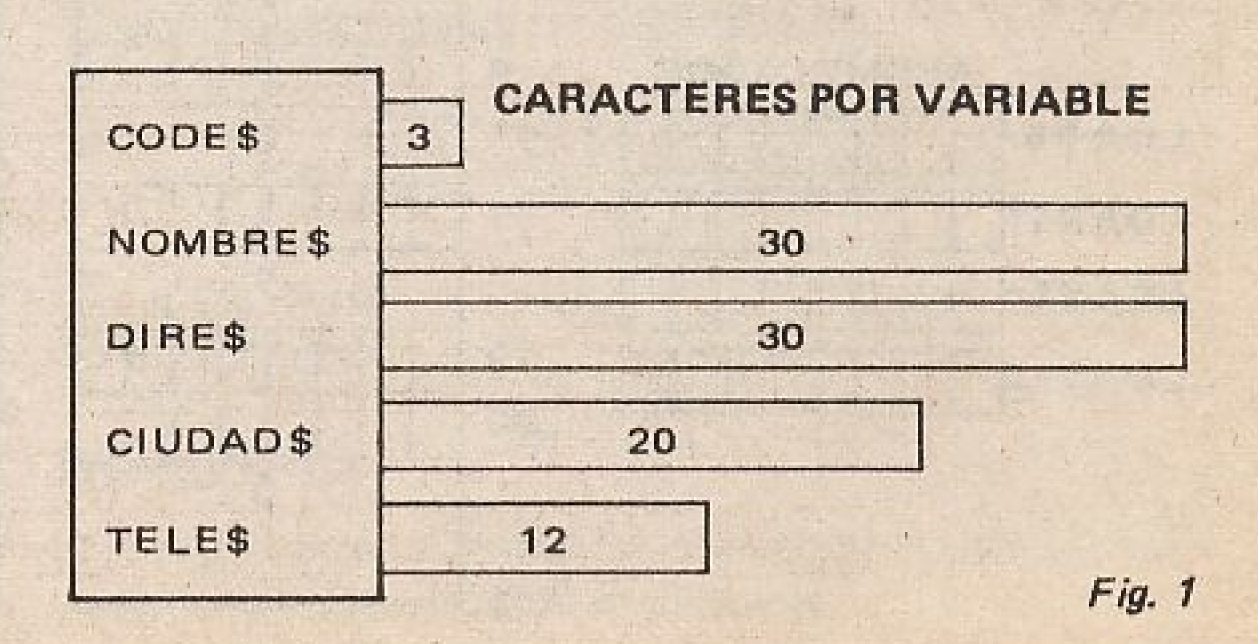
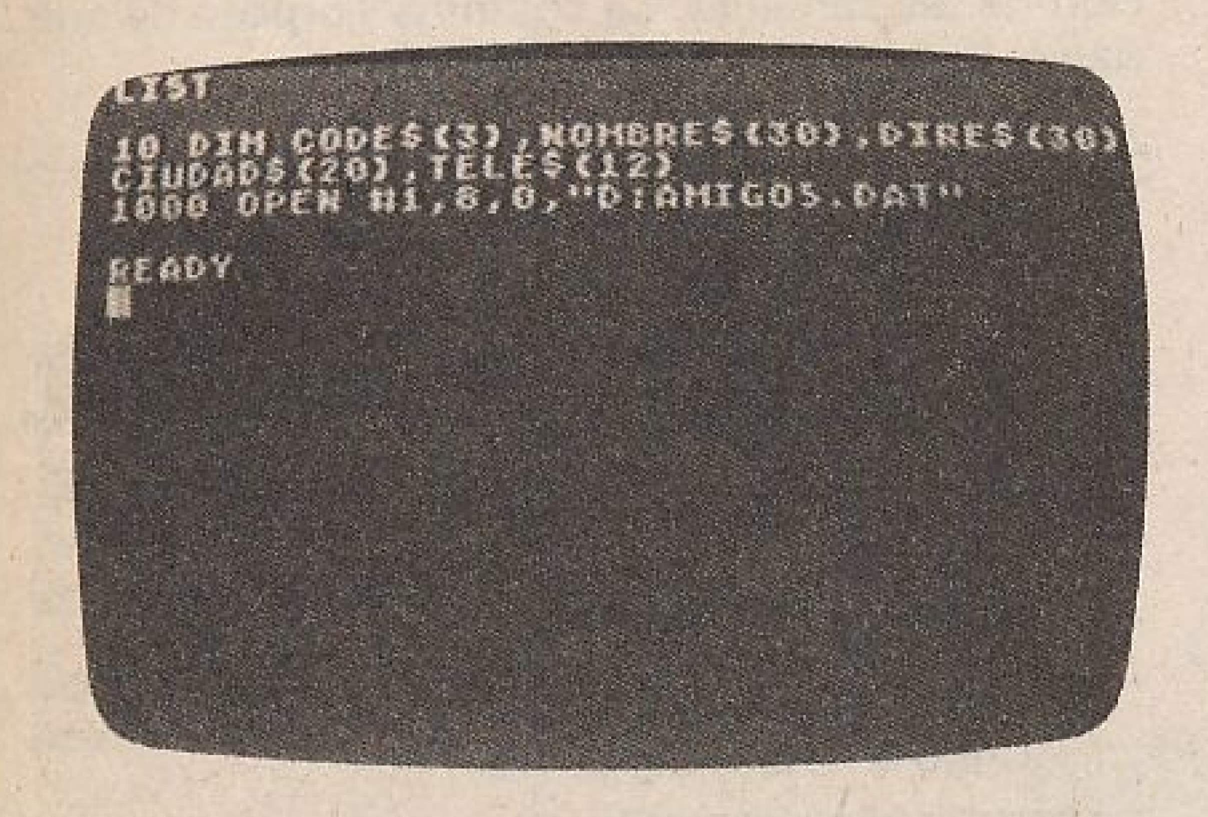


Fig. 1

La línea siguiente indica lo afirmado literalmente:

```
10 DIM CODE$(3),NOMBRE$(30),DIRE$(30),
CIUDAD$(20),TELE$(12)
```

Ingrese el programa ARCHI.PRG con el comando ENTER. Borre la pantalla y liste, la foto ilustra su presentación:



4. Ingresar un registro al archivo.

La instrucción PRINT # IOCB permite escribir directamente hacia el archivo del directorio utilizando el número del block abierto para ese fin.

Para ingresar en la variable alfanumérica utilice la instrucción INPUT.

Las líneas siguientes indican lo afirmado:

```
2000 REM INGRESA UN REGISTRO
2010 GRAPHICS 0
2020 PRINT "INGRESA EL CODIGO ":INPUT
CODE$:PRINT #1;CODE$
2030 PRINT "INGRESA EL NOMBRE ":INPUT
NOMBRE$:PRINT #1;NOMBRE$
2040 PRINT "INGRESA DIRECCION ":INPUT
DIRE$:PRINT #1;DIRE$
2050 PRINT "INGRESA CIUDAD ":INPUT CIU
DAD$:PRINT #1;CIUDAD$
2060 PRINT "INGRESA TELEFONO ":INPUT T
ELE$:PRINT #1;TELE$
```

Ejecute el programa e ingrese los datos de la foto de la siguiente pantalla. Ellos corresponden a los del suscriptor No. 774 a nuestra revista MUNDOATARI, el cual por el hecho de ser mencionado en esta columna ha recibido un presente, como una muestra de reconocimiento para nuestros suscriptores.

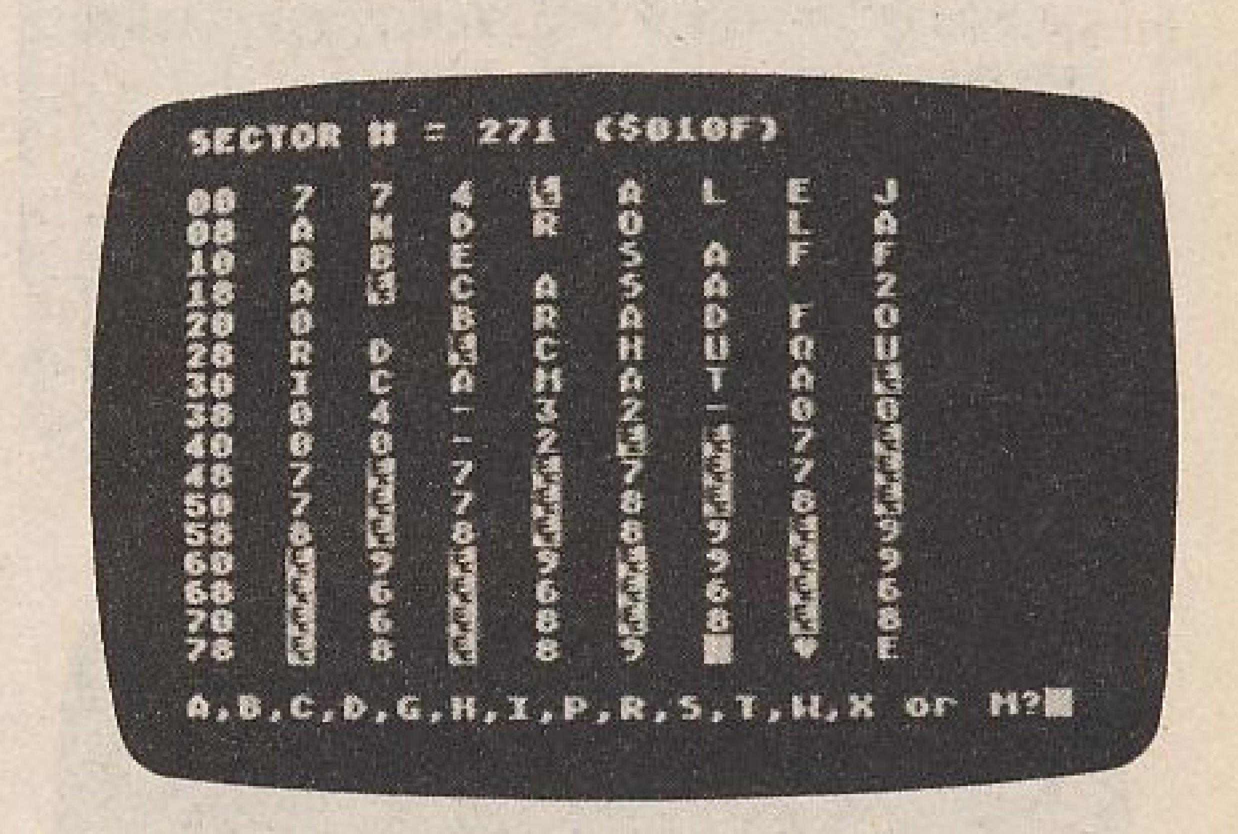
¿Cómo quedó almacenada esta información en el diskette?

En la figura 2, a pie de página, se muestra la estructura del archivo en diskette.

Problemas para el próximo mes:

- Agregue lo necesario para ingresar 10 registros en este programa.
- ¿Cuál es la rutina para el proceso inverso, o sea, traer los amigos hacia la memoria de su ATARI?
- ¿Cómo ver esta información en el diskette? Los examinadores de diskette, programas comerciales americanos permiten ver el contenido de los sectores del diskette.
  La foto ilustra el uso de uno de ellos: el DISCAN, examinando el contenido del sector, en que se almacenó la información. Observe el símbolo que se registra cuando se presiona la tecla RETURN.

Hasta el próximo mes.



ESQUEMA DE UN REGISTRO EN DISKETTE

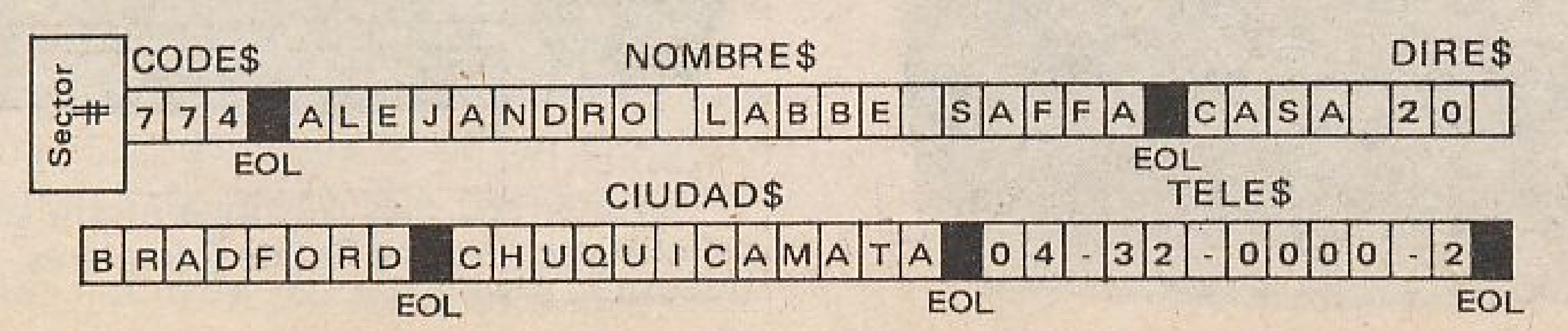


Fig. 2

De byte en byte

La columna **De Byte en Byte** finaliza este mes el bloque de localizaciones que controlan la pantalla.

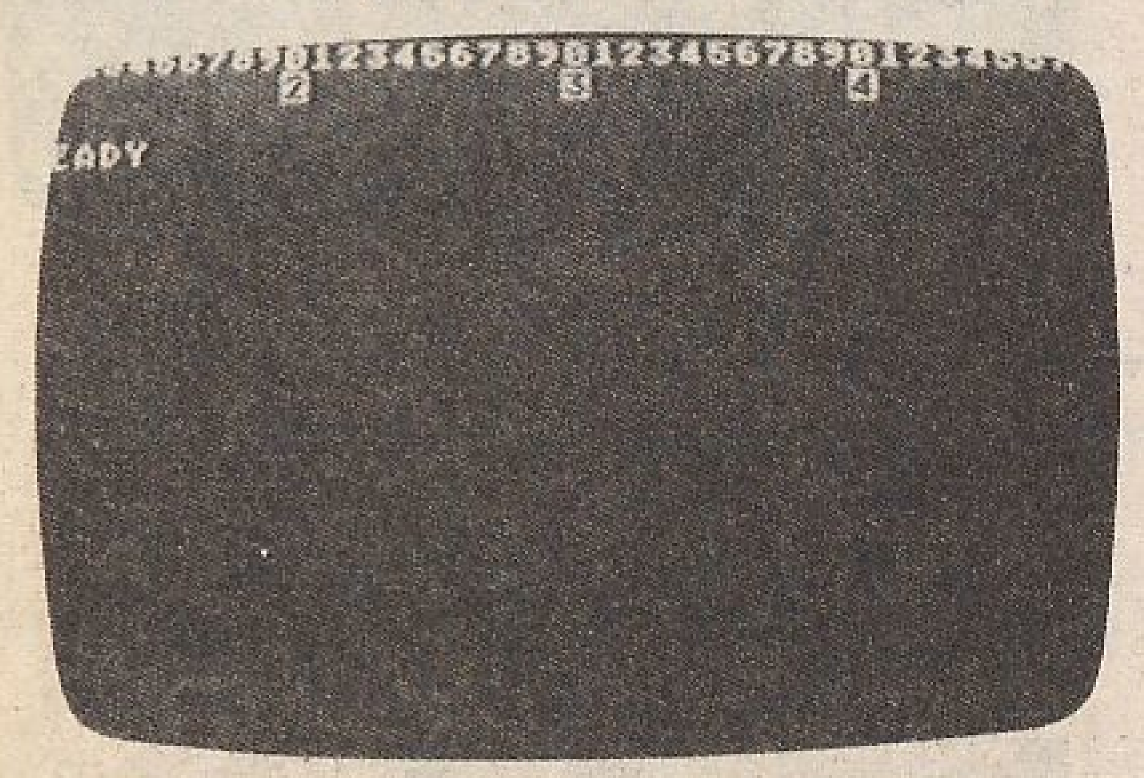
10. NUMERO DE ESPACIOS DEL TABULADOR
Localización 201

Digite el listado siguiente:

```
10 GRAPHICS 0:POKE 82,0
20 POSITION 0,0:? "0123456789012345678
90123456789012345678 90"
30 POKE 84,1
40 PRINT "1","2","3","4"
```

Ejecute y observe en pantalla la relación entre el operador coma (,) y la posición de los números en la pantalla.

La foto de pantalla ilustra lo afirmado.



Digite en modo directo:

```
PRINT PEEK(201)
```

Agregue la línea:

```
35 POKE 201,5
```

y compare el resultado en relación al anterior. Utilice esta localización para optimizar la presentación de las pantallas. La columna **Torpedo** del mes siguiente aplicará esta localización.

11. PRESENCIA DEL CURSOR
Localización 752

Durante la ejecución de programas comerciales es un efecto optimizador la eliminación del cursor.

Digite en modo directo:

```
10 PRINT PEEK(752)
```

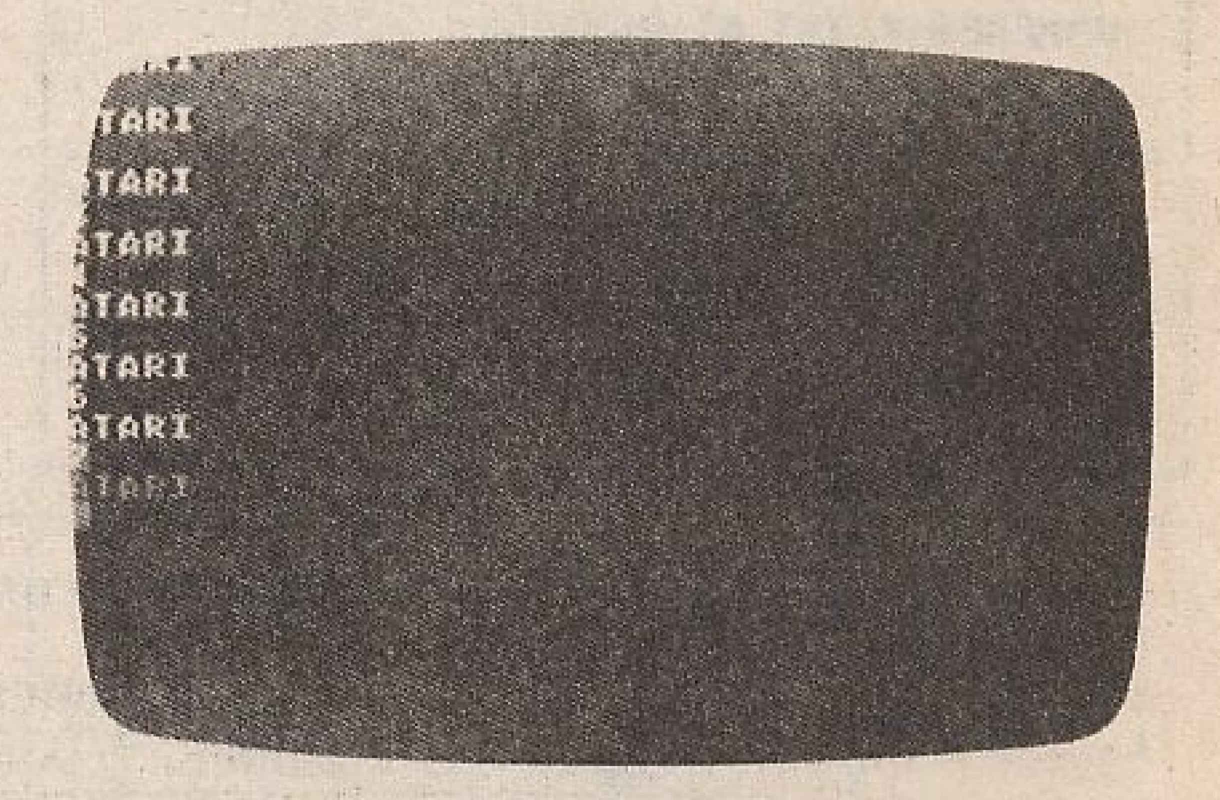
Ejecute y examine el valor de retorno.

Digite el listado siguiente:

```
10 GRAPHICS 0
20 FOR J=1 TO 255
30 POKE 752,0
35 PRINT "ATARI"
40 A=1^1^1
50 POKE 752,J
55 PRINT J
60 A=1^1^1
70 NEXT J
```

Observe que valores enteros mayores que 0 hasta el 255 hacen el cursor invisible.

Elimine la línea 30 del programa y compare las ejecuciones.



El próximo mes continuará esta columna con las localizaciones del **Set de Caracteres**.

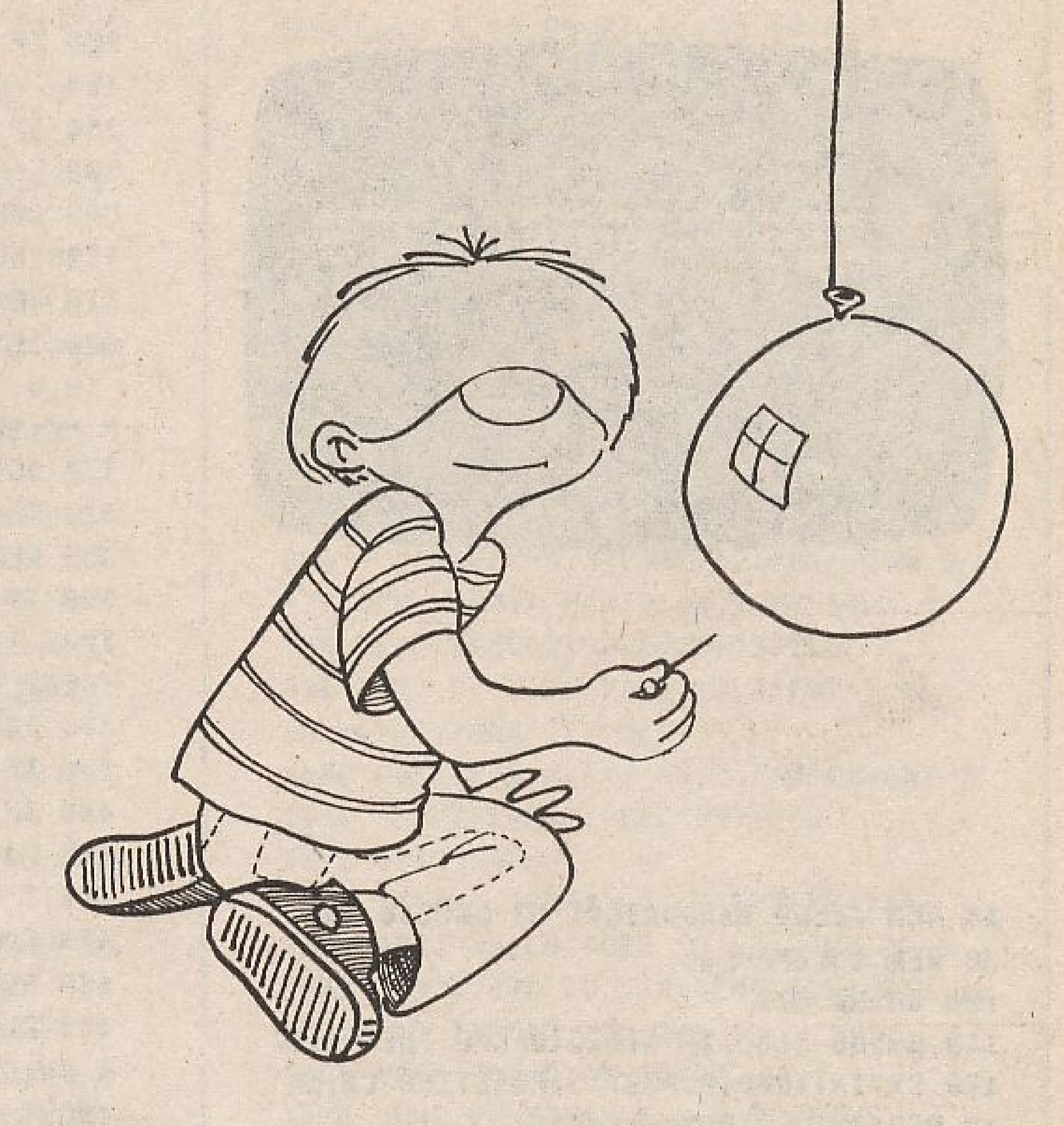
Hasta entonces. ●

JUEGO DEL MES:

Carambola

Cada mes publicaremos un juego no bélico, que desarrolle ciertas habilidades.

Se aceptan contribuciones de nuestros lectores



La columna JUEGO DEL MES hace un llamado a todos los fans de la animación a crear y diseñar programas simples para ser publicados en esta sección.

El juego de este mes es simple. Digite el listado correspondiente y grábelo en su periférico de memoria permanente.

Para iniciar apriete el botón rojo del joystick. Usted dispone de 3 figuras para eliminar con un toque las frutas que se encuentran en la parte superior de la pantalla. Cuando la figura choca con ellas se produce un rebote y desciende. En este momento debe mover usted la barra horizontal que es controlada por el joystick para hacer CARAMBOLA. El trabajo del jugador consiste en hacer rebotar permanentemente la figura hasta terminar con las frutas colgadas.

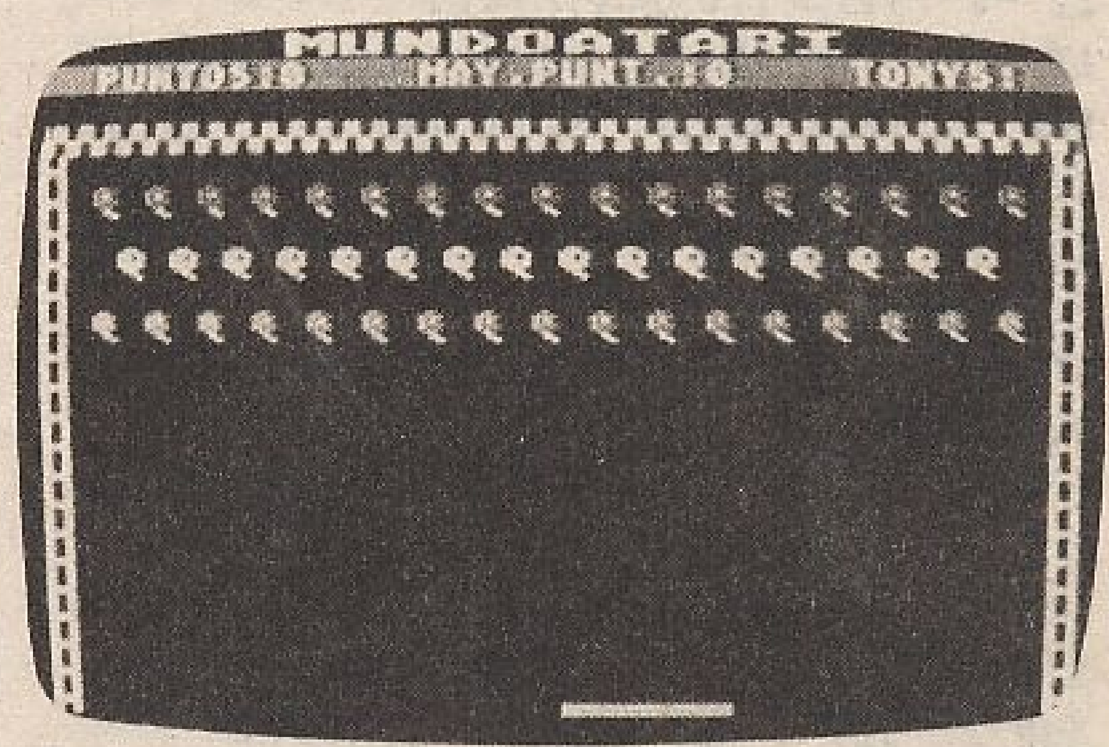
Las fotos siguientes ilustran las situaciones descritas:

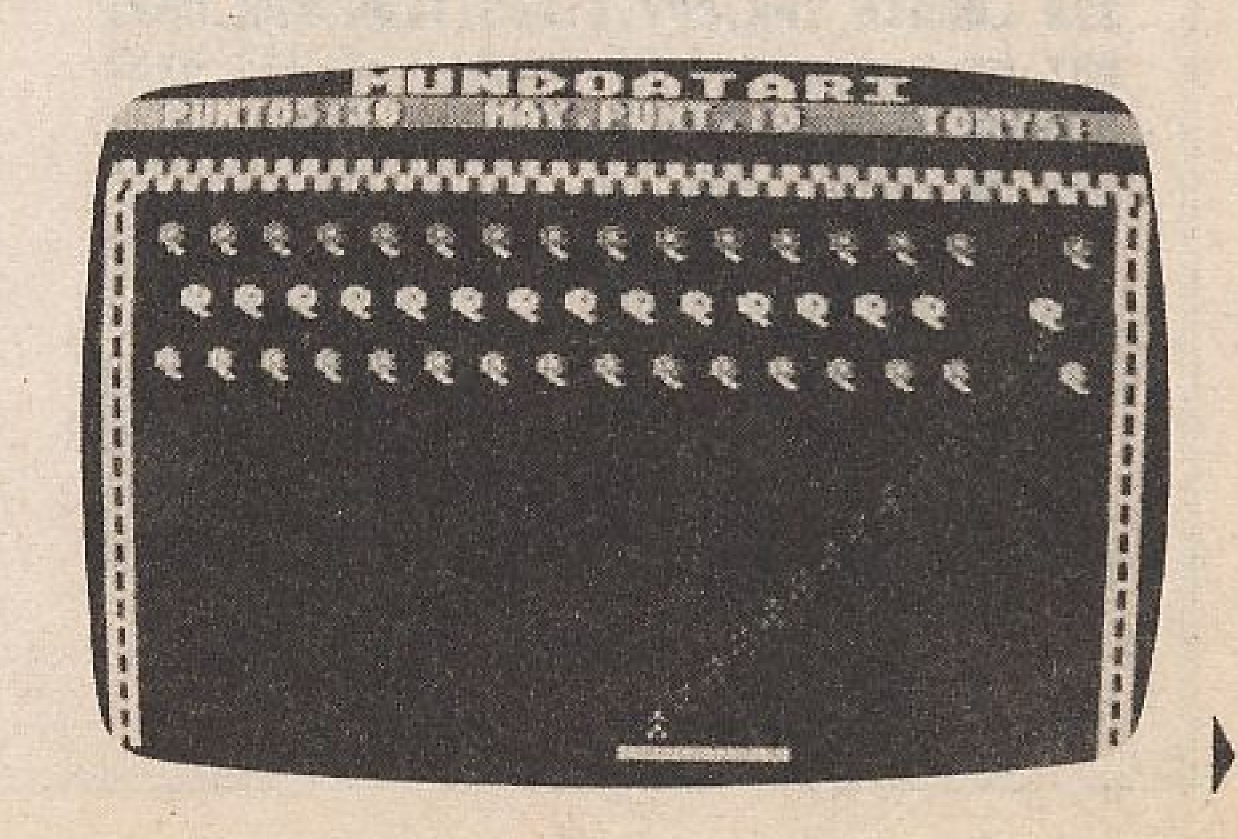
En el programa JUEGO DEL MES se aplican algunos conceptos de Lista de Despliegue (ver al respecto columna TECNICAS).

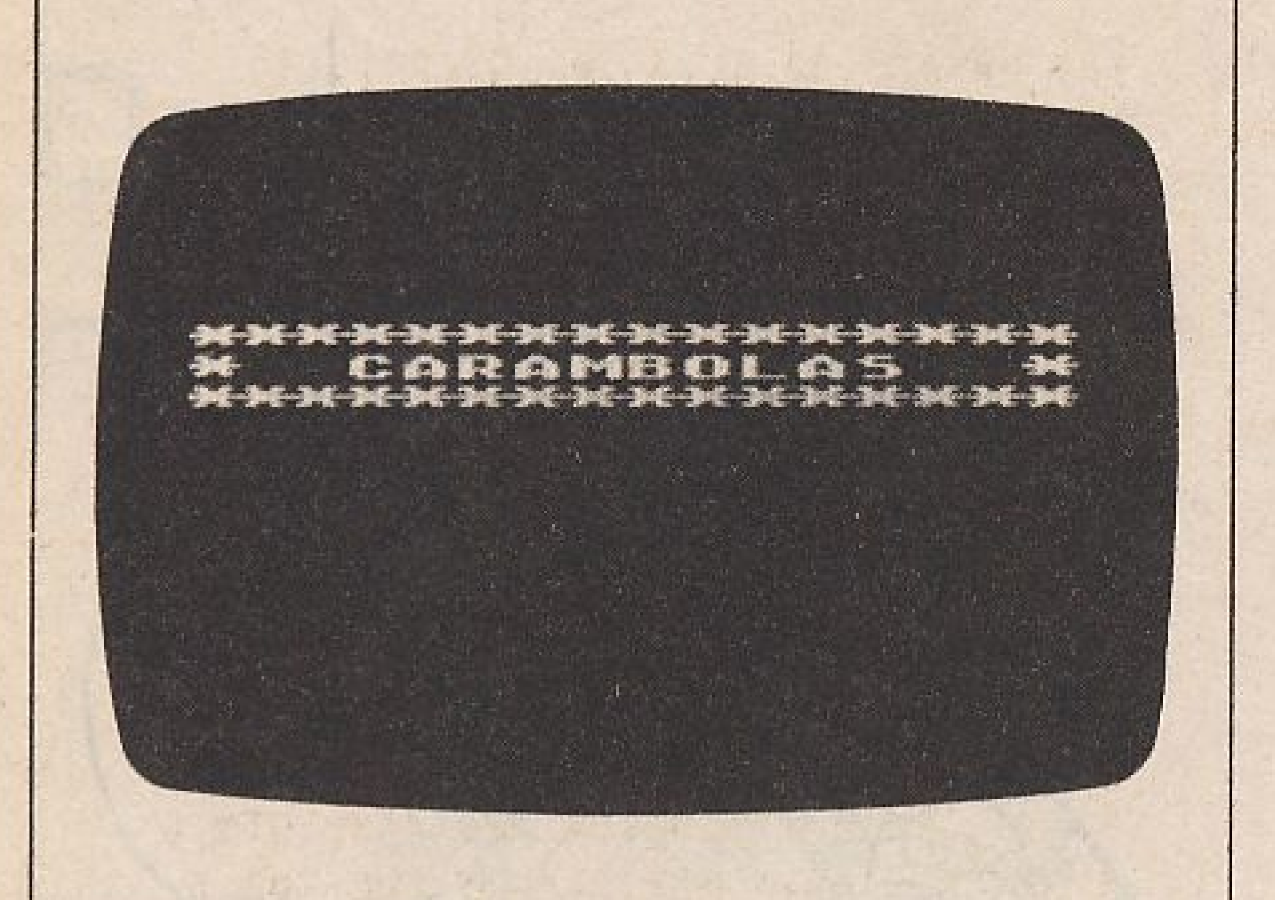
Examine las líneas 980 a 1070 y escríbanos sobre la interpretación de estas líneas con los conceptos de instrucciones ANTIC y punteros.

Recuerde que las cartas se envían a nombre de la columna a la casilla 458, Correo Ñuñoa, Santiago.

Hay premios para las soluciones correctas.



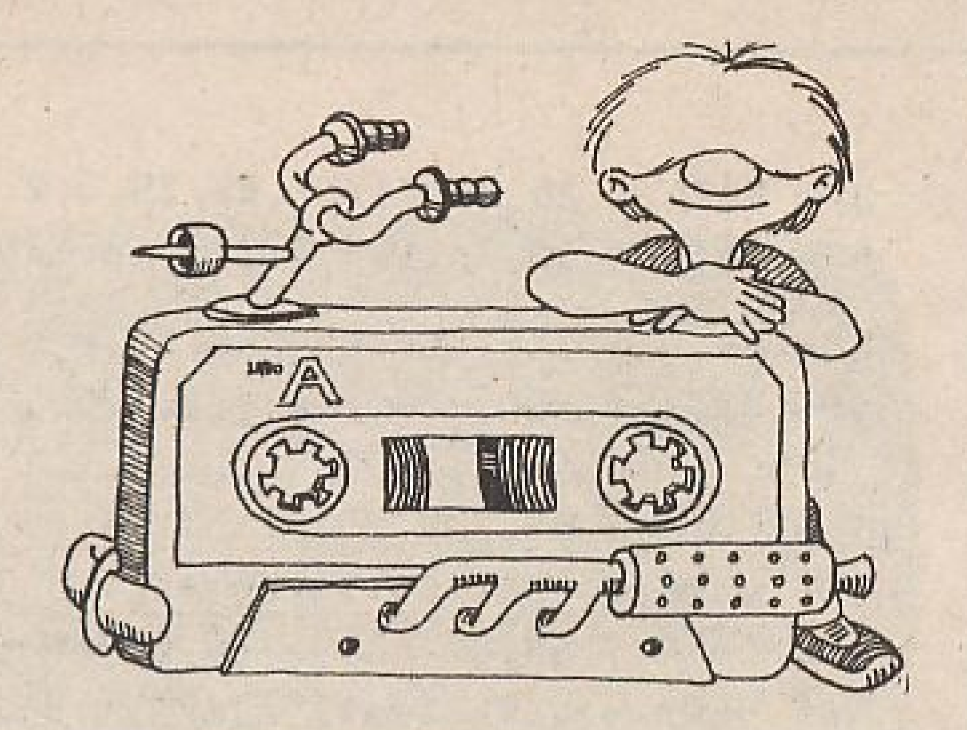





```
10 REM JUEGO MUNDOATARI #3 AGOSTO 1987
60 REM CARAMBOLAS
100 GOSUB 620
110 GOSUB 1180:IF STRIG(0)<>0 THEN 110
120 CX=INT(RND(0)*32)+5:POSITION CX,15
:? "$":CYI=-1:CXI=1:CY=15
130 REM *** RUTINA DE LECTURA PALANCA
140 S=STICK(0):IF S<>11 AND S<>7 OR S=
15 THEN 200
150 IF S=7 THEN NP=PP+2:CH=1:PAD$=PADR
$:IF NP>ED THEN NP=ED
160 IF S=11 THEN NP=PP-2:CH=-1:PAD$=PA
DL$:IF NP<2 THEN NP=2
170 IF S<>LS THEN POSITION PP,21:? PAD
$:LS=S:GOTO 190
180 LS=S:POSITION NP,21:? PAD$:PP=NP
190 IF RND(0)>0.85 THEN CH=0
200 IF SET THEN SET= NOT SET:CYI=-1
210 CPY=CY+CYI:CPX=CX+CXI
220 LOCATE CPX,CPY,Z:POSITION CPX,CPY:
PUT #6,Z
230 IF CY>21 THEN 440
240 IF Z=32 THEN 410
250 IF Z<>37 AND Z<>38 AND Z<>32 AND Z
<>169 THEN 320
260 FOR I=15 TO 0 STEP -5:SOUND 0,0,2,
I:NEXT I
270 IF CX=36 AND CY>3 THEN CXI=-1:CPX=
CPX-INC:GOTO 380
280 IF CX=2 AND CY<21 THEN CXI=1:GOTO
380
290 IF CY=3 THEN CYI=1:GOTO 380
300 IF CY=20 THEN CYI=-1:CXI=CH:INC=1:
CPX=CPX+INC:POSITION CX,CY:? " ":CX=CX
+INC:GOTO 380
310 REM *RUTINA ESTALLIDO GLOBOS*
320 SET=1:CXI=CXI*-1:SC=SC+IN:POSITION
 10,0:? SC:BAL=BAL-1:FOR I=16 TO 0 STE
P -4:SOUND 1,50,0,I:NEXT I
330 SC2=SC2+IN
340 IF BAL=7 THEN 530
350 REM * RUTINA  EXTRA *
360 IF SC2>490 THEN SC2=0:CL=CL+1:FOR
I2=1 TO 5:SOUND X,10,10,8:FOR I=1 TO 5
:NEXT I:SOUND 0,0,0,0:NEXT I2
370 POSITION 38,0:? CL:GOTO 410
380 IF CPY>20 OR CPY<3 THEN CPY=CY
400 IF CPX>36 OR CPX<2 THEN CPX=CX
410 POSITION CX,CY:? " ":POSITION CPX,
CPY:? "$":CX=CPX:CY=CPY
420 GOTO 140
430 REM *RUTINA CAIDA  *
440 CL=CL-1:POSITION 38,0:? CL:POSITIO
N CX,CY:? "(":FOR T2=1 TO 3:GOSUB 1140
:NEXT T2:SOUND 0,0,0,0:FOR A=0 TO 35
450 POSITION A,22:? " QB":FOR T2=1 TO
17:NEXT T2:NEXT A:POSITION 36,22:? "
":IF CL<>0 THEN 110
460 REM * RUTINA FIN DEL JUEGO **
470 GOSUB 580:POSITION 0,1:FOR X=8 TO
3 STEP -2:FOR I=3 TO 36:SOUND 0,I,10,8
:POSITION I,X:? " ":NEXT I:NEXT X
480 SOUND 0,0,0,0:REM POKE 53279,0
490 FOR I=15 TO 36:POSITION I,0:? "  "
:NEXT I
500 IF PEEK(53279)<>6 AND STRIG(0)<>0
THEN POSITION 15,0:? "FIN JUEGO! CARGA
 START  ":GOSUB 590:GOTO 500
510 SOUND 0,0,0,0:SOUND 1,0,0,0:GOSUB
1070:GOTO 110
520 REM * RUTINA LADO DE LAS RAQUETAS
530 TIM=TIM+1:FOR IZ=3 TO 36:POSITION
IZ,21:? " ":NEXT IZ
540 IF TIM=1 THEN PADR$="  %%%%":ED=31
:PADL$="%%%%  ":IN=30:GOTO 560
550 IF TIM=2 THEN PADR$="  %%":ED=33:P
ADL$="%%  ":IN=50
560 POSITION CX,CY:? " "
```

```
570 POSITION PP,21:? CL2$:GOSUB 1100:G
OTO 110
580 FOR IZ=3 TO 36:POSITION IZ,21:? "
":NEXT IZ:RETURN
590 SOUND 0,200,10,8:SOUND 1,199,10,8
600 RETURN
610 REM * RUTINA TITULOS *
620 GRAPHICS 1+16
625 POSITION 2,7:? #6;"***************
*"
630 POSITION 2,8:? #6;"*  CARAMBOLAS
*"
635 POSITION 2,9:? #6;"***************
*"
670 REM * VARIABLES DE INICIALIZAC.*
680 DIM ROW1$(34),ROW2$(34),ROW3$(34),
TP$(39),SD$(60),PAD$(8),CL$(8),CL2$(6)
:POKE 82,0
690 DIM PADR$(8),PADL$(8):FOR I=1 TO 3
2 STEP 2:ROW1$(I,I+1)="# ":NEXT I
700 FOR I=1 TO 34 STEP 2:ROW2$(I)=CHR$
(39):ROW2$(I+1)=CHR$(32):NEXT I
710 FOR I=1 TO 34 STEP 2:ROW3$(I)=CHR$
(167):ROW3$(I+1)=CHR$(32):NEXT I
720 FOR I=1 TO 37:TP$(I,I)="&":NEXT I
730 FOR I=1 TO 57 STEP 3:SD$(I)=CHR$(2
9):SD$(I+1)=CHR$(30):SD$(I+2)=CHR$(169
):NEXT I
740 PADR$="  %%%%%%":PADL$="%%%%%%  ":
PAD$=PADR$
750 FOR I=1 TO 8:CL$(I,I)=" ":NEXT I
760 CL=3:SC=0
770 REM * CARACTERES EN LENGUAJE MAQ*
780 FOR M=0 TO 59:READ Q:POKE 1536+M,Q
:NEXT M
790 DATA 169,0,133,205,133,203,133,207
,169,224,133,206,104,104,104,133,204,2
4,165,207,105,1,133,207,201,5
800 DATA 240,26,160,0,177,205,145,203,
200,208,249,24,165,204,105,1,133,204,2
4,165,206,105,1,133,206,76
810 DATA 17,06,165,208,141,244,2,96
820 POKE 559,0:POKE 106,PEEK(106)-5:ST
CHR=(PEEK(106)+1)*256
830 C=STCHR/256:D=USR(1536,C)
840 REM * REDEFINIR CARACTERES *
850 FOR COUNT=1 TO 9:READ OLD:NC=OLD*8
860 FOR CHANGE=0 TO 7:READ NUM:POKE ST
CHR+CHANGE+NC,NUM:NEXT CHANGE:NEXT COU
NT
870 DATA 3,20,85,81,85,85,20,8,2
880 DATA 4,32,32,184,48,16,16,136,136
890 DATA 5,0,0,255,85,255,85,0,0
900 DATA 6,250,250,250,250,175,175,175
,175
910 DATA 7,60,255,251,255,255,60,8,2
920 DATA 8,136,136,16,16,48,184,32,32
930 DATA 9,65,65,65,85,85,65,65,65
940 DATA 49,170,174,191,174,170,170,48
,48
950 DATA 34,168,168,130,130,170,170,14
,12
960 ? :POKE 756,STCHR/256:POKE 559,0
970 REM * DATA PANTALLA Y MUESTRA *
980 DL=PEEK(560)+PEEK(561)*256
990 POKE DL+3,70:POKE 87,1:FOR I=DL+7
TO DL+29:POKE I,4:NEXT I
1000 POSITION 4,0:? #6;" MUNDOATARI "
1010 SS=PEEK(88)+PEEK(89)*256
1020 SCR=SS+20
1030 HI=INT(SCR/256):LO=SCR-HI*256
1040 POKE 88,LO:POKE 89,HI:POKE 87,0
1050 FOR I=708 TO 712:READ COL:POKE I,
COL:NEXT I:POKE 752,1
1060 DATA 26,10,68,134,0
1070 CL=3:POSITION 3,0:? "PUNTOS:
MAY.PUNT.:      TONYS: ":IF SC>HS THEN
HS=SC
1080 TIM=0:IN=10:SC2=0:ED=29:SC=0:POSI
TION 1,2:? TP$:POSITION 2,2:? SD$:POSI
TION 38,2:? SD$
1090 PADR$="  %%%%%%":PADL$="%%%%%%  "
:PAD$=PADR$
1100 POKE 559,34:POSITION 3,4:? ROW2$:
POSITION 4,6:? ROW1$:POSITION 3,8:? RO
W3$
1110 PP=18:POSITION PP,21:? PADR$
1120 BAL=50:POSITION 10,0:? SC:POSITIO
N 25,0:? HS:POSITION 38,0:? CL
1130 RETURN
1140 FOR I4=110 TO 30 STEP -5
1150 SOUND 0,I4,10,8:FOR T=1 TO 2:NEXT
 T:NEXT I4
1160 FOR I4=30 TO 110 STEP 5:SOUND 0,I
4,10,8:FOR T=1 TO 2:NEXT T:NEXT I4:RET
URN
1180 XX=PEEK(16):IF XX<128 THEN 1220
1200 POKE 16,XX-128:POKE 53774,XX-128
1220 RETURN
```

●



Controlando la casetera

SOLUCION AL NUMERO ANTERIOR

En realidad no interesa saber el nombre de la variable con que fue grabada la información.
El siguiente listado lee del casete y edita el contenido en pantalla:

```
5 OPEN #1, 4, 0, "C:"
10 TRAP 30
15 GET #1, C
20 ? CHR$(C);
25 GOTO 15
30 CLOSE #1
```

La instrucción de línea 15 toma los bytes del casete una y otra vez hasta llegar al último, el fin de archivo (EOF), con el que se produce un error 136. Pero ese error estaba previsto en la línea 10 y el programa es enviado a la línea 30.
Si usted quiere determinar la cantidad de datos grabados, sólo debe ubicar un contador que controle las veces que el programa pasa por la línea 15. Pero esa solución la dejamos en sus manos.

En el número anterior tratamos el uso de PUT # y GET #, como alternativas de PRINT # e INPUT # para grabar y cargar datos en el casete.

Luego analizamos el caso de la limitación de INPUT para recuperar más de 254 bytes.

De los ejemplos tratados y del problema comentado más arriba podemos deducir que estos comandos son perfectamente compatibles: si grabamos datos con PRINT # podemos recuperarlos con GET #.

DEL TECLADO AL CASETE

A continuación vamos a experimentar con la grabación de datos a medida que se digitan.

Para esto usaremos 2 canales de comunicación: uno para el teclado (K:) y otro para la casetera (C:).

Se recomienda revisar los conceptos contenidos en MUNDOATARI No. 0, pág. 31, relativos a la "entrada inteligente por el teclado", para entender mejor nuestro tema de hoy y para apreciar las ventajas de esta técnica en cuanto a aplicar restricciones en el ingreso.

Manos al teclado. Nuestra misión por ahora se trata de grabar su nombre directamente desde el teclado. Digite las siguientes líneas:

```
10 OPEN #1,4,0,"K:"
15 OPEN #2,8,0,"C:"
20 GET #1,N
25 PUT #2,N
30 CLOSE #1:CLOSE #2
```

Comentarios del listado:

10 Abre el canal 1 para entrada desde el teclado. Cada tecla presionada produce el valor ASCII correspondiente
15 Abre el canal 2 para grabar en la casetera
20 Saca el valor de la tecla digitada y lo guarda en la variable N
25 Pone ese byte en la casetera

Ejecute con RUN y observe el comportamiento:

— la casetera toca 2 beeps para advertirle que ponga la cinta en posición y presione las teclas RECORD y PLAY
— al oprimir una tecla se inicia la sincronización por aproximadamente 20 segundos
— el computador espera que usted oprima una tecla para ... ¿qué pasa?, ¡no responde ni se detiene!... ¡ALTO! ...

Oprima BREAK y repasemos qué anduvo mal. Luego de terminar el período de inicio (línea 15) el computador espera el primer ingreso y la casera no se detiene hasta que no haya sido ejecutado ese requerimiento.

Una forma práctica de controlar el problema es poner como primera información un **registro vacío,** o sea, un registro que contenga ceros, espacios o cualquier otro byte. Después de esto la casetera mantendrá un buen comportamiento.

Un registro vacío se ingresa como sigue:

```
FOR X=0 TO 127: PUT # 2, 0: NEXT X
```

Está claro que ese registro debe ser considerado posteriormente al recuperar la información. Normalmente es descartado.

BUFFER

Hagamos un pequeño paréntesis. Aprovecharemos de conversar acerca de un detalle que más de algún lector lo ha advertido.

En el programa inicial de esta columna se ordenaba al computador sacar un byte e imprimirlo. Sin embargo, en la ejecución sucedía algo distinto: el computador rescata una serie de bytes y de pronto los imprime todos juntos; luego rescata otra serie y los imprime a continuación; y así sucesivamente hasta terminar.

¿Problema de temperamento? No, es sólo un trabajo con economía de esfuerzo.

El computador no graba o rescata datos individualmente, sino que usa como recurso el **buffer**, sector de la memoria donde acumula temporalmente los datos hasta que se llena. Recién entonces procesa la orden siguiente, mientras mantiene detenido el motor de la casetera (tiempo de IRG).

El buffer está ubicado entre las localizaciones 1021 y la 1151, pero los datos propiamente tales se almacenan a partir de la localización 1024.

Ahora volvamos atrás y trabajemos en serio. A continuación ingresaremos desde el teclado un texto para grabar en casete y observaremos el comportamiento del buffer a medida que se completa. Digite para esto el listado siguiente:

```
1 REM **Graba directo del teclado**
5 GRAPHICS 0:BUF=1024
6 ? "RETROCEDA LA CINTA":? "Y PRESIONE
 RECORD+PLAY":?
9 PRINT " PRESIONE Return"
10 OPEN #1,4,0,"K:"
15 OPEN #2,8,0,"C:"
16 ? CHR$(125):? "DIGITE EL TEXTO..."
20 FOR X=0 TO 127:PUT #2,0:NEXT X
25 GET #1,N
30 IF N=27 THEN 60
35 IF C>127 THEN C=0:? :?
40 PUT #2,N
45 ? CHR$(PEEK(BUF+C));
50 C=C+1
55 GOTO 25
60 CLOSE #1:CLOSE #2:END
```

El registro vacío está en la línea 20. La variable BUF indica el comienzo del buffer, el cual en la línea 45 es incrementado con el contador C, para indicar su contenido. La línea 30 espera la tecla ESC para terminar el ingreso. La línea 35 pone el contador en 0 una vez que llega a 128.

Ejecute el listado con RUN. Una vez que la casetera se detiene después de la rutina de inicio, digite cualquier texto, incluya errores y correcciones, haga uso de los caracteres gráficos, minúsculas, inversos, etc.

Para terminar digite la tecla ESC.

¿Terminó todo? Muy bien, grabe su programa en casete con la instrucción LIST "C:", apague el computador y veamos que resultó de lo anterior.

Ahora digite el siguiente listado de rescate:

```
65 REM **Carga grabacion a memoria *
66 GRAPHICS 0
67 PRINT " RETROCEDA LA CINTA"
68 PRINT " PRESIONE PLAY"
69 PRINT :PRINT "PRESIONE Return"
70 OPEN #1,4,0,"C:"
75 FOR X=0 TO 127:GET #1,A:NEXT X
80 TRAP 110
85 GET #1,N
95 ? CHR$(N);
100 FOR T=1 TO 20:NEXT T
105 GOTO 85
110 CLOSE #1:END
```

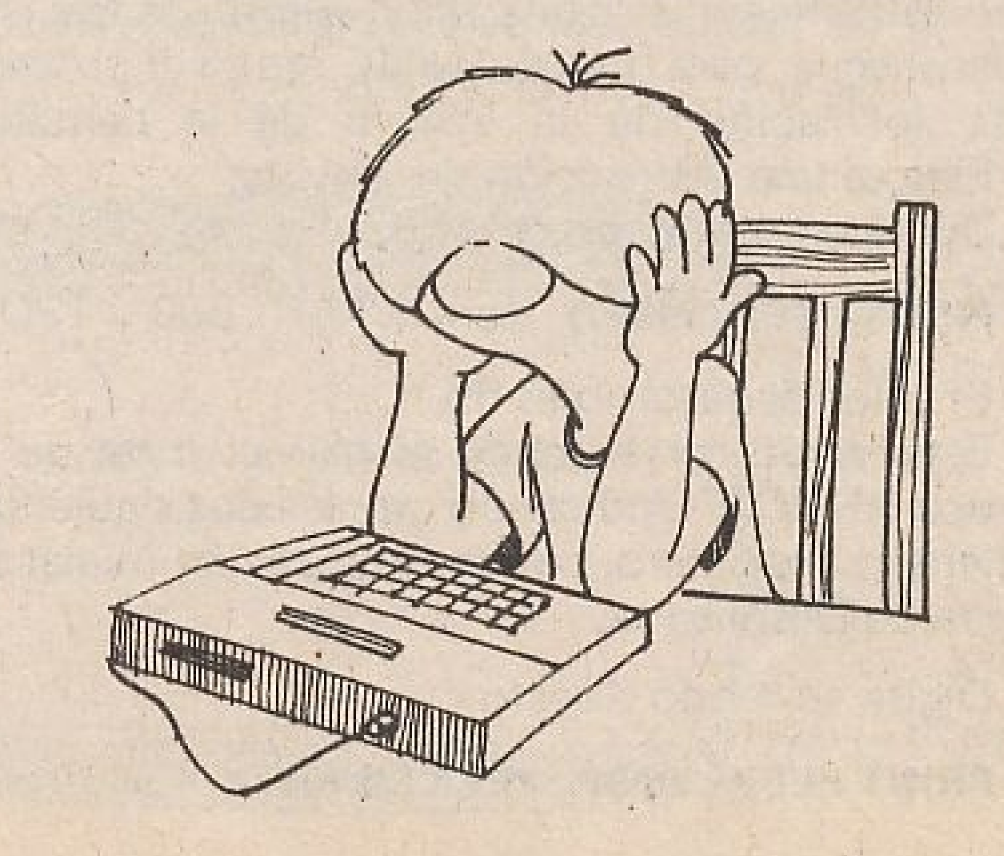
Comentarios del listado:

75 Rescate del registro vacío y pérdida de él
80 Al producirse un error por EOF se cierra el canal y termina el programa
100 Ciclo de demora para ver el rescate de datos en cámara lenta.

Ejecute el programa con RUN y observe su desarrollo. Si lo prefiere puede eliminar la línea 100 para tener la velocidad normal de la operación.

PROBLEMA PARA EL PROXIMO NUMERO:

El problema pendiente consistirá en optimizar la rutina de grabación para que la digitación sea continua. Elimine la representación del buffer y reemplácela por una edición directa a pantalla.



Lista de Despliegue

Este número de MUNDOATARI finaliza la técnica de Lista de Despliegue, demostrando su utilidad en la columna PROYECTO de este mes.

INSTRUCCIONES JUMP

El ANTIC recorre las instrucciones de la Lista de Despliegue que corresponden a líneas (ANTIC y LMS), varias veces por segundo.

Si la capacidad de formar imágenes del ojo humano fuera más rápida, podría captar los destellos de la pantalla. La cámara fotográfica es capaz de captar estos fenómenos en la pantalla utilizando un tiempo corto de exposición. Ver foto que ilustra el efecto mencionado.



Ahora bien, al finalizar la Lista de Despliegue se encuentra la instrucción JUMP, que indica un nuevo salto hacia la localización inicial de la Lista de Despliegue, para que el ANTIC repita el proceso de la formación de la imagen de la pantalla.

Esta es una instrucción de 3 bytes:

Digite en modo directo:

PRINT PEEK(39997) para ATARI 800 y 130

El valor de retorno es 65.

Este valor corresponde al primer byte de la instrucción JUMP, indicando que los dos siguientes contienen un puntero para la localización inicial de la Lista de Despliegue.

Digite en modo directo:

PRINT PEEK(39998), PEEK(39999)

Para el ATARI 800 y 130 XE los valores a retornar son: 32 y 156.

¿Qué sentido tienen estos números?.

Ellos indican una dirección. Para comprobar digite:

PRINT 32 + 256 * 156

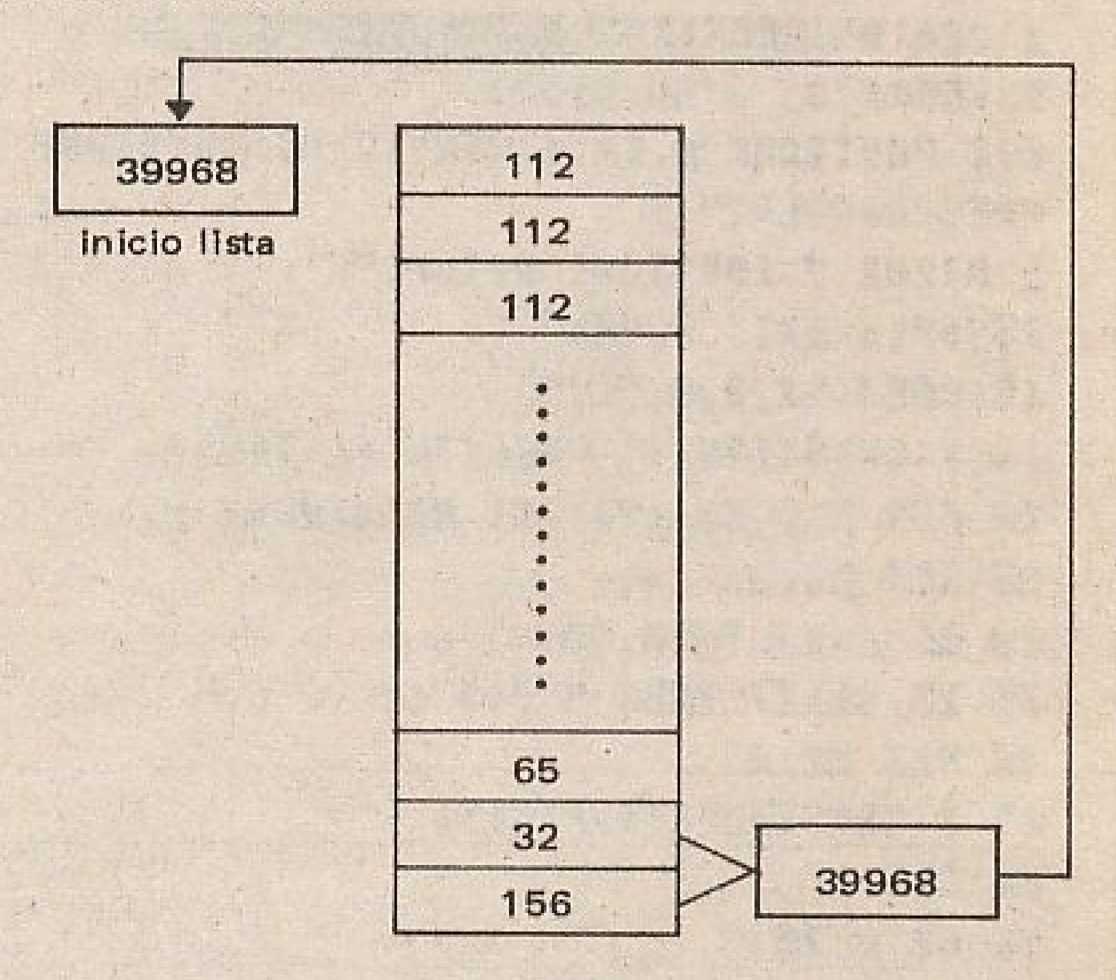
El valor de retorno es: 39968.

Compare con el valor que resulta de:

PRINT PEEK(560) + 256 * PEEK(561)

Es lo mismo. Si aún persisten dudas al respecto, revise la columna TECNICAS de MUNDOATARI Nos. 0, 1 y 2, en relación a este mismo tema.

El esquema siguiente muestra el direccionamiento para el microprocesador ANTIC de la instrucción JUMP.



RESUMEN DE INSTRUCCIONES:

1. LINEAS BLANK SCAN
   Son instrucciones que indican al ANTIC no desplegar datos en la pantalla. El valor es variable.
2. LMS
   Instrucción de 3 bytes que permite desplegar una línea específica del Modo ANTIC y direccionando a un área específica de la memoria para leer los datos a editar en pantalla.

3. ANTIC
Instrucción que el microprocesador ANTIC interpreta según el valor de dicha instrucción.
4. JUMP
Instrucción de 3 bytes que permite repetir continuamente el proceso de despliegue de la pantalla por el microprocesador ANTIC.

A continuación aplicaremos este resumen para una Lista de Despliegue específica.

Digite el siguiente listado:

```
10 REM PROGRAMA LISTA DESPLIEGUE
15 REM MUNDO ATARI 3 AGOSTO 87
20 GRAPHICS 31:POKE 82,0
30 INILISTA=PEEK(560)+256*PEEK(561)
35 PRINT "LOC INICIAL LISTA = ";INILIS
TA
40 PRINT " INSTRUCCIONES FORMATO LINEA
S SCANS"
50 FOR J=0 TO 2:PRINT PEEK(INILISTA+J)
;"-";:NEXT J
60 POSITION 0,4:? " INSTRUCCION LMS (3
 BYTES )"
70 FOR J=3 TO 5:PRINT PEEK(INILISTA+J)
;"-";:NEXT J
75 POSITION 0,6:? "INSTRUCCIONES ANTIC
80 FOR J=6 TO 198
90 PRINT PEEK(INILISTA+J);"-";
95 NEXT J
100 POSITION 0,22:? "INSTRUCCIONES JUM
P "
110 FOR J=199 TO 201:? PEEK(INILISTA+J
);"-";:NEXT J
120 GOTO 120
```

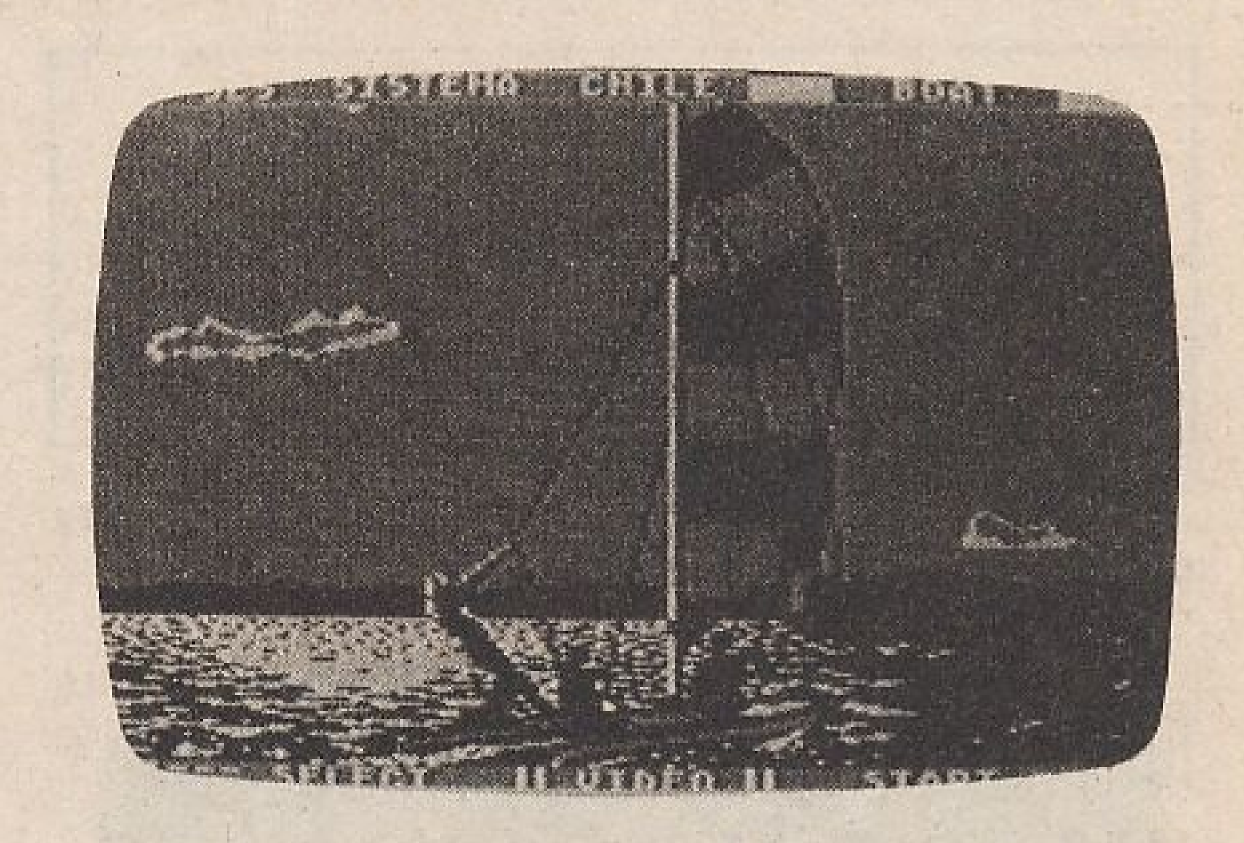
Ejecute y revise los conceptos del resumen con el ejemplo de la pantalla.

Ahora una aplicación, ocupando como base esta Lista de Despliegue Modo 7 + o ANTIC 14.

PROBLEMA:

Usted tiene pantallas que ha creado con Tablero de Dibujo CX-77 (ver en MUNDOATARI No. 2) y desea incorporarlos en pantalla, pero necesita agregar un texto superior en Modo 0 para colocar su nombre, y otro en la parte inferior como medio de comunicación con el usuario.

La foto de pantalla ilustra lo afirmado:



Para realizar esta pantalla primero se necesita un conocimiento preciso de la Lista de Despliegue del Modo Gráfico 15 (31).

La ejecución del listado 1 entrega este listado, pero para una mejor ilustración compare con el esquema a pie de página.

La primera columna contiene el valor de la localización de memoria específica de ubicación de la Lista (localización 32822 al 33023).

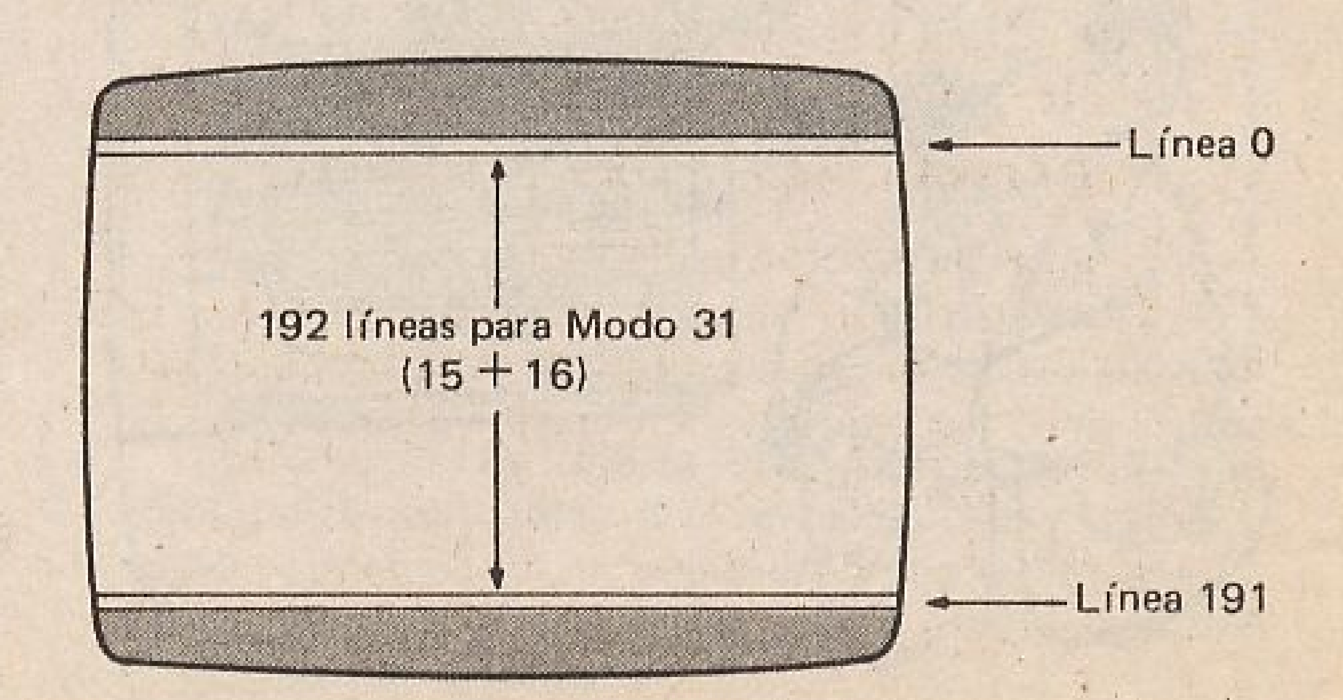
La segunda columna indica el valor contenido en esta localización, los que corresponden a los analizados para otros casos en números anteriores.

Se interrumpe con líneas de puntos para conectar físicamente con un esquema de la ubicación física de instrucción con línea de pantalla para el modo gráfico en uso.

Los textos explican el tipo de instrucción y las interacciones de las mismas.

Loc. Mem.	Lista Despl.	Comentarios
32822	112	24 líneas scan
:	112	
	112	
32825	78	Define modo lista (64 + 14)
32826	80	LMS
32827	129	
32828	14	194 instrucciones ANTIC
:	:	
33021	65	Jump
33022	54	Apunta al inicio de la Lista de Despliegue
33023	128	

LISTA DE DESPLIEGUE MODO 15 (7 +) ANTIC 14



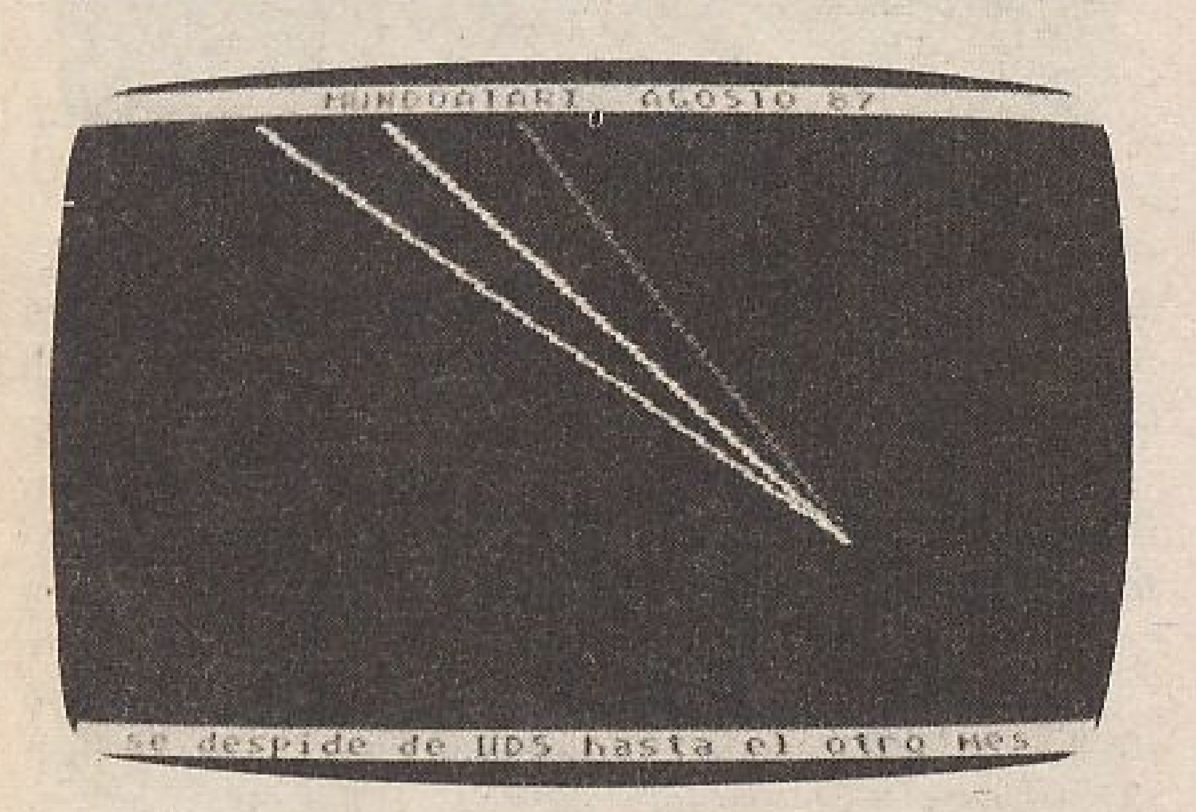
Ahora digite el siguiente listado:

```
10 GRAPHICS 31
20 POKE 32828,2
30 GOTO 30
```

Ejecute y observe la pantalla.
Agregue las líneas siguientes:

```
30 FOR X=0 TO 4
40 COLOR X
50 PLOT 10+X*20,5:DRAWTO 120,120
60 NEXT X
100 GOTO 100
```

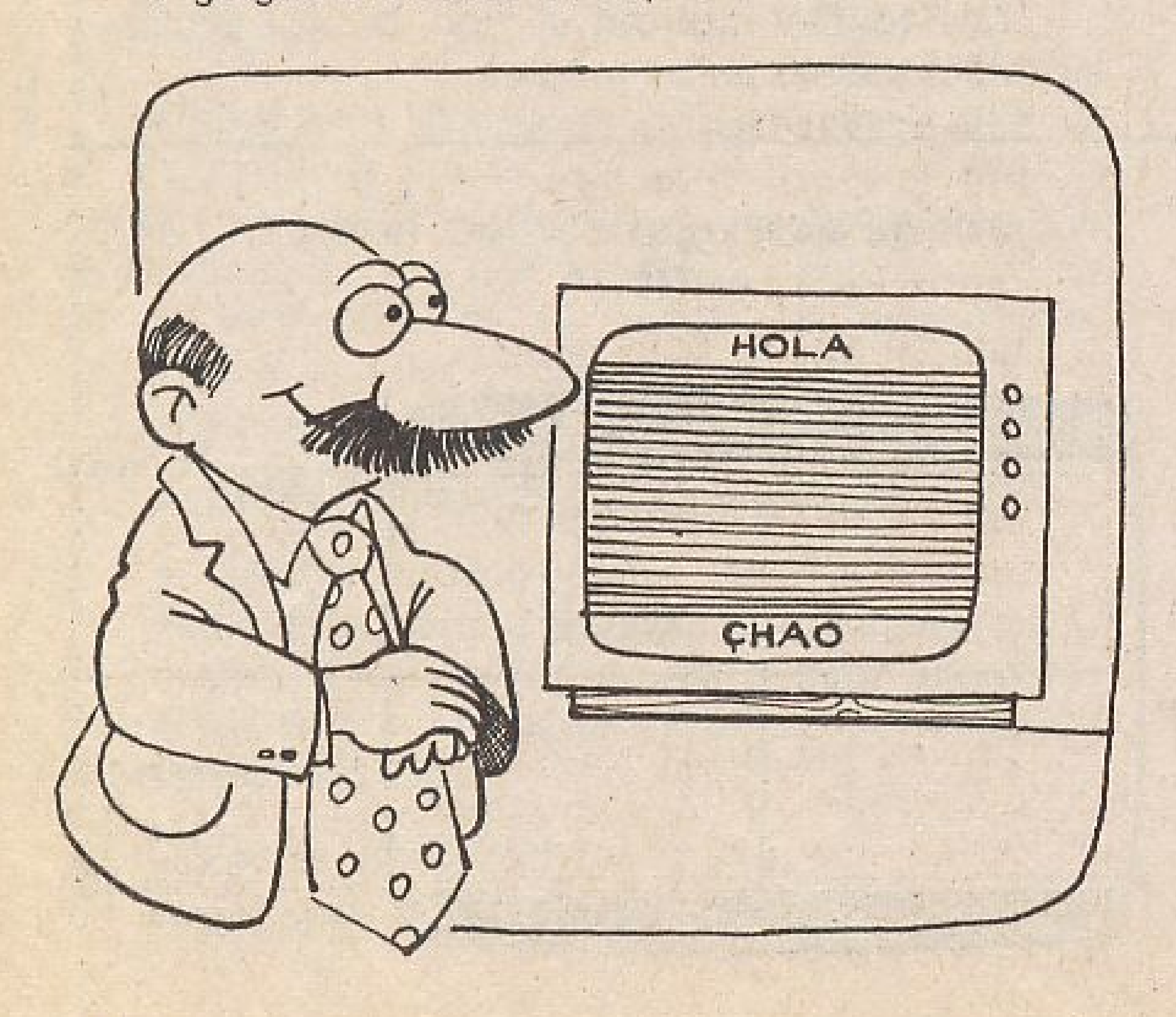
Ejecute y observe la pantalla. La foto ilustra el aspecto de la modificación.



¿Cómo introducir ahora un texto en la parte superior en Modo 0?

Para ello recurriremos a conceptos analizados en MUNDOATARI No. 1 en la columna DE BYTE EN BYTE.

Agregue el listado de la próxima columna:




```
100 POKE 87,0:POKE 88,80:POKE 89,129
110 POSITION 0,1
120 ? "          MUNDOATARI  AGOSTO 87          ";
130 GOTO 130
```

Ejecute y observe el resultado.

Expliquemos a continuación los valores insertados en la línea 100.

Los valores 80 y 129 son los valores de la instrucción LMS, que direcciona hacia la región de la memoria para extraer los datos (ver esquema de localizaciones 32826 y 32827).

Hagamos una prueba de la importancia de la instrucción POSITION para este truco.

Modifique la línea 110 por:

POSITION 0, 0

Ejecute y compare el resultado. Es absolutamente necesario el conocimiento y comprensión de la Lista de Despliegue.

Para la línea final agregue las líneas faltantes del listado siguiente:

```
10 GRAPHICS 31:POKE 82,0
20 POKE 32828,2:POKE 33004,2
30 FOR X=0 TO 4
40 COLOR X
50 PLOT 10+X*20,5:DRAWTO 120,120
60 NEXT X
100 POKE 87,0:POKE 88,80:POKE 89,129
110 POSITION 0,1
120 ? "          MUNDOATARI  AGOSTO 87          ";
130 POKE 87,0:POKE 88,80:POKE 89,156
140 POSITION 0,2:? " se despide de UDS hasta el otro mes      ";
150 GOTO 150
```

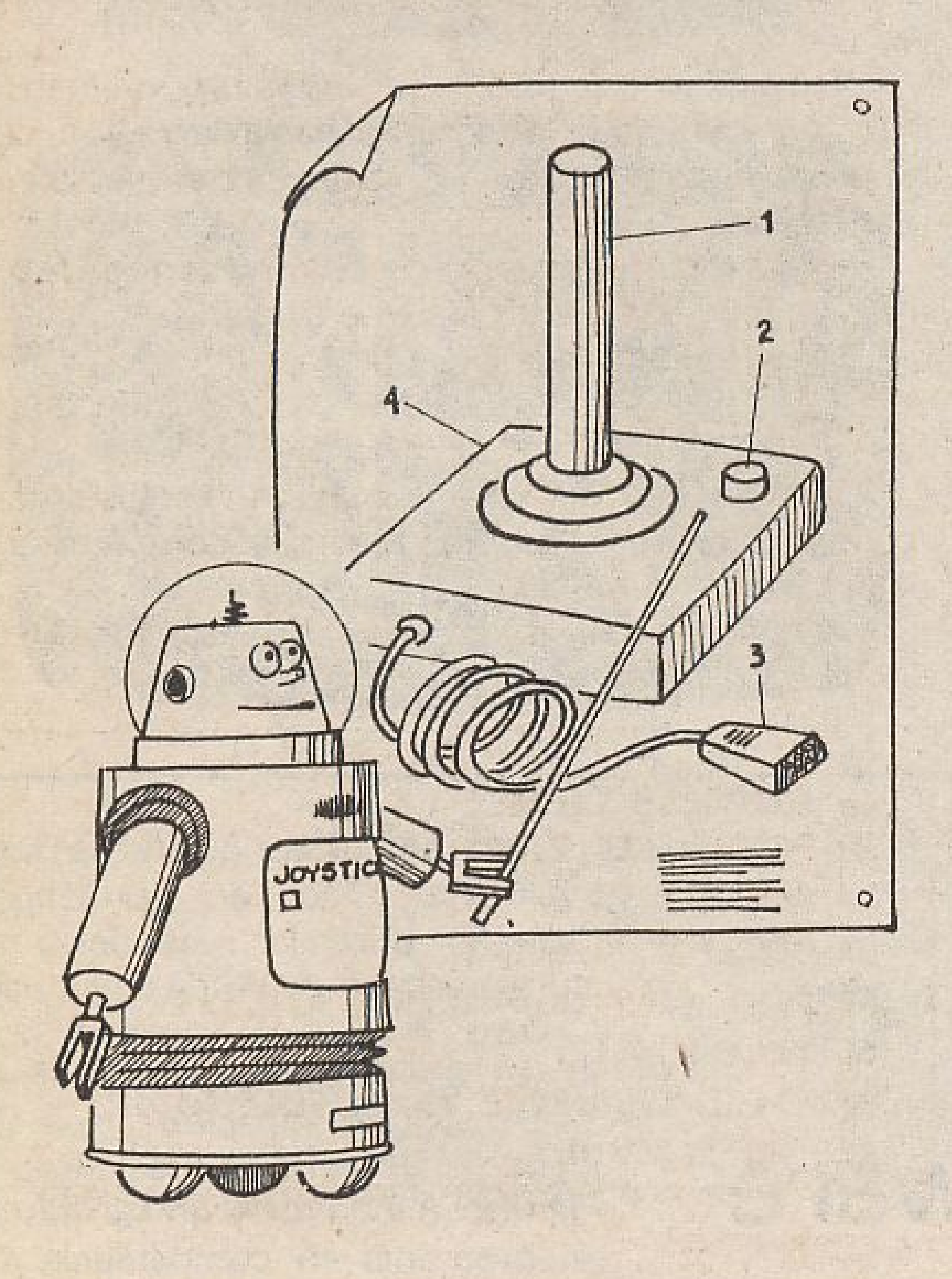
Ejecute y disfrute del mensaje.

Al parecer el problema se encuentra resuelto, pero no en su totalidad, pues quedan pendientes unas interrogantes para que usted las resuelva. Dirija su carta a casilla 458, Correo Ñuñoa, indicando el nombre de la columna.

1. ¿Cuál es la razón para pokear el valor 2 en la localización 33004?
2. Explique los valores de la línea 130.
3. Modifique los valores de estas líneas y obtenga el mismo resultado.

La columna CLASES de este mes aplica conceptos de Lista de Despliegue en la salida de datos a pantalla. Profundice sus conocimientos con ese ejemplo práctico.

Primeros pasos en ATARI BASIC



La palabra **joystick** se asocia inmediatamente al uso de videojuegos, pero programandolo adecuadamente puede tener usos variados y así por ejemplo, el efecto de destreza motriz de la primera edad puede agilizarse en forma considerable.

La rutina coleccionable de este mes proporciona información sobre el manejo del joystick y de la instrucción STICK del ATARI BASIC.

Digite las siguientes líneas de programa:

```
10 REM PROG1 ATARIBASIC AGOSTO 87
20 PRINT STICK(0)
30 A=1^1^1^1
40 GOTO 10
```

Inserte el joystick en la entrada 1 y ejecute.

Observe los valores que se editan en la pantalla. Cambie la dirección de la palanca del joystick y compare los valores con los del esquema siguiente:

10	14	6
11	15	7
9	13	5

¿Cómo examinar los valores para la entrada 2 del joystick? Modifique el valor 0 de la línea 20 por el 1 y cambie el joystick a la entrada 2.

Ejecute nuevamente y compare con los valores anteriores.

Con estos conceptos preparatorios pasaremos a la rutina de este mes para dibujar con el joystick en modo gráfico 4.

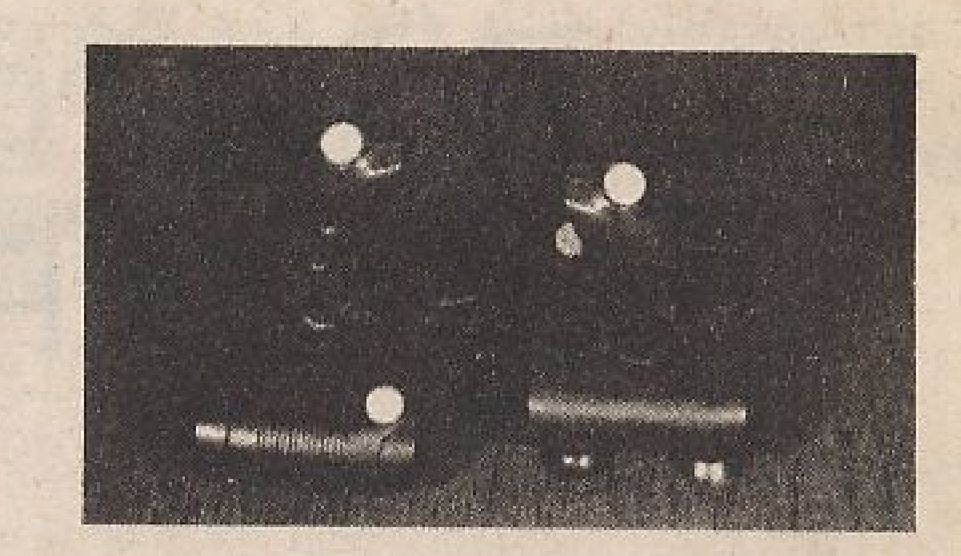
```
10 GRAPHICS 20
20 DIM JX(15),JY(15)
30 FOR I=5 TO 15
40 READ X,Y
50 JX(I)=X:JY(I)=Y
60 NEXT I
70 DATA 1,1,1,-1,1,0,0,0,-1,1,-1,-1,-1
,0,0,0,0,1,0,-1,0,0
100 PX=40:PY=20
110 COLOR 1
120 PLOT PX,PY
130 S=STICK(0)
140 IF S=15 THEN 110
170 PX=PX+JX(S)
180 PY=PY+JY(S)
200 IF PX>79 THEN PX=0
210 IF PY>39 THEN PY=0
220 IF PY<0 THEN PY=39
230 IF PX<0 THEN PX=79
240 GOTO 110
```

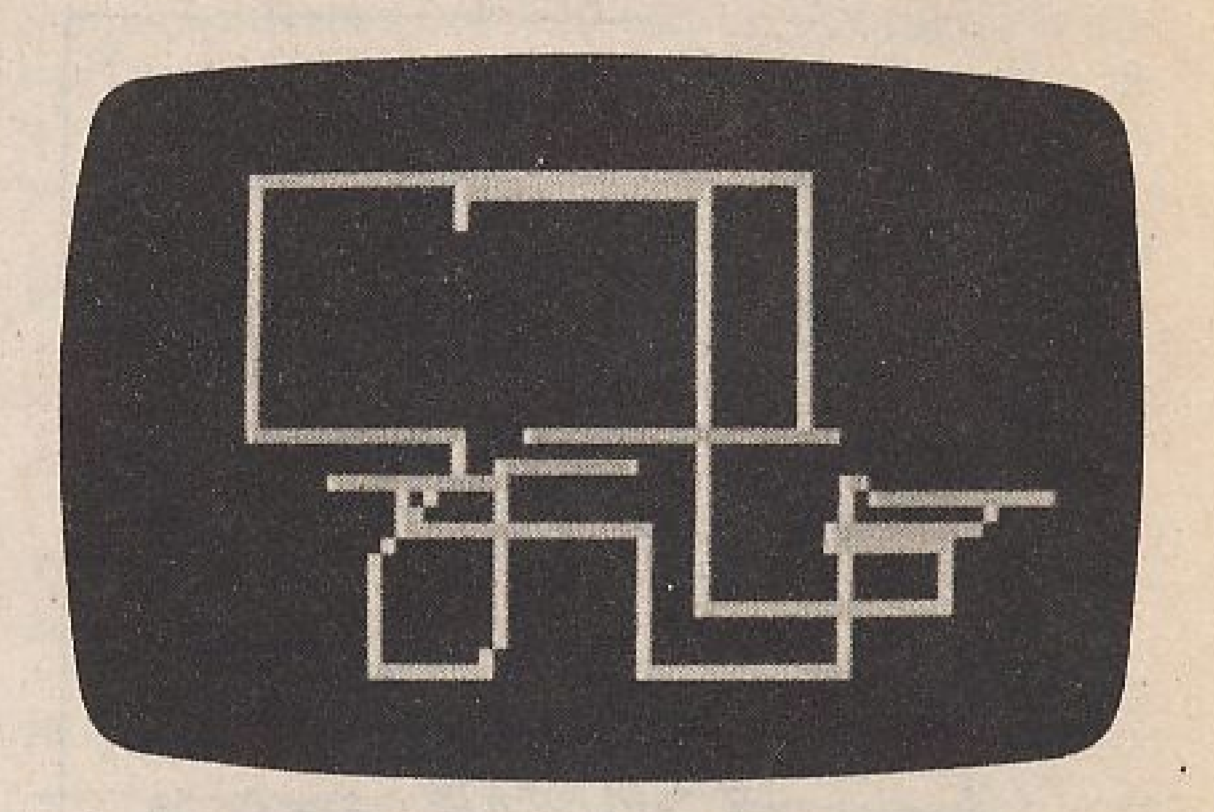
Esta rutina de dibujo, asociada con el joystick como entrada, utiliza dos arreglos: JX y JY, los cuales incrementan la posición de variable para la nueva posición direccionada con el movimiento de la palanca del joystick.

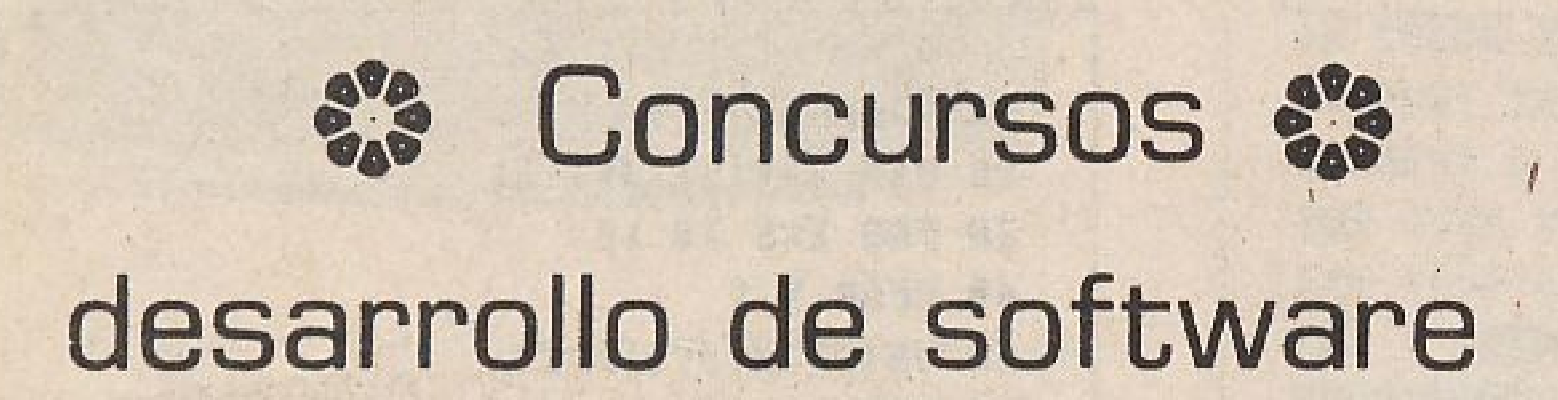
El siguiente cuadro proporciona la relación de incrementos:

Joystick		Arreglos	
Dirección	Valor	JX	JY
↘	5	1	1
↗	6	1	–1
→	7	1	0
	8	0	0
↙	9	–1	1
↖	10	–1	–1
←	11	–1	0
	12	0	0
↓	13	0	1
↑	14	0	–1
	15	0	0

Esta rutina tiene aplicaciones prácticas. La columna PEQUES de este mes la utiliza para su programa "dibujando el uno".







Concursos desarrollo de software

ATARI CLUB de Chile ha organizado para el presente año, la realización de dos concursos de desarrollo de software de creación propia. El primero orientado al nivel de usuario, en el cual podrán participar todos los socios del Club, enviando al concurso un programa ATARI de creación original, desarrollado en cualquiera de los lenguajes de programación disponibles para el computador (BASIC, LOGO, ASSEMBLER, etc.). El programa en competencia puede ser de cualquier naturaleza o aplicación, siendo aconsejable que pertenezca a una de las siguientes categorías:

- **Entretención y desarrollo personal,**
- **Educación general,**
- **Aplicaciones profesionales, o**
- **Software utilitario.**

El segundo concurso está dirigido exclusivamente a colegios poseedores de Talleres computacionales ATARI y que sean miembros del Club. Los programas participantes a nivel de colegios deberán ser sólo de carácter educacional. Los profesores y alumnos del colegio en conjunto, podrán desarrollar un programa para hacer participar a su establecimiento educacional en esta competencia.

Para ambos concursos, los socios interesados en participar, deberán retirar las bases del concurso respectivas desde cualquier Centro ATARI a lo largo del país, las cuales incluyen el "Registro de participación" que se deberá completar con los antecedentes solicitados y enviar junto al programa en competencia a:

ATARI CLUB DE CHILE
Concurso Desarrollo de Software
Av. Vicuña Mackenna 1705
Santiago

o entregarlos personalmente en cualquiera de los Centros ATARI.

La fecha final de recepción de los programas en competencia, se ha fijado para el día:

30 de octubre de 1987

Los resultador de los socios ganadores, serán dados a conocer a través del diario El Mercurio de Santiago el día 5 de diciembre de 1987.

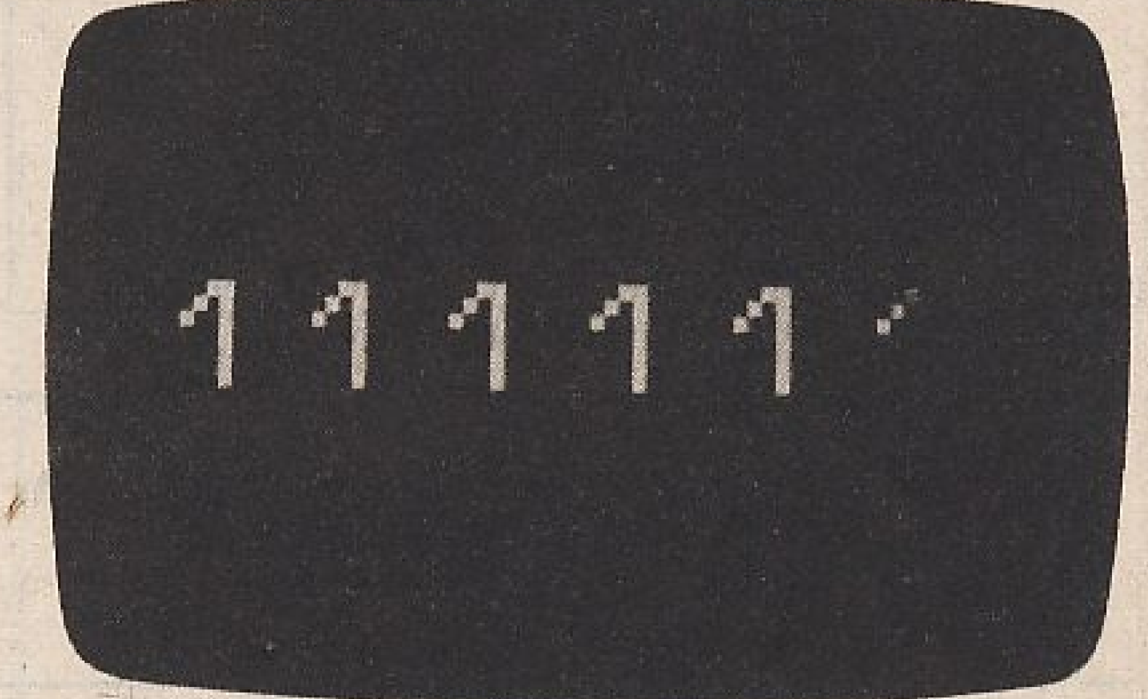
Hay considerados excelentes premios para los socios ganadores de ambos concursos.

¡Participe sin demora!

EDUCANDO CON ATARI

PEQUES:

Dibujando el uno



¡Qué típico resulta decir acerca de nuestros pequeños, que se encuentran aburridos y no saben qué hacer!. La foto de nuestra introducción ilustra el problema de miles de padres en relación al interés de sus hijos.

La generación actual, ya en su vida intrauterina recibe información tecnológica que fue reservada para nuestras últimas décadas.

Por otro lado, el tiempo de permanencia frente al televisor y la forma de presentar juegos y otros de realidad concreta provocan desaliento en los pequeños, cosa que nuestra generación no sufría.

Ahora el ATARI puede resolver en parte esta problemática. La creación de programas simples pero **interactivos** es una necesidad que debe estimular en cierta medida a los que formamos la gran familia atariana.

Este mes, con el programa **Dibujando el Uno,** damos un esquema para construir programas simples, que con el uso y participación de los pequeños se incrementará, y quien sabe si pasen a promoverse en forma masiva. Usted, amigo lector, tiene la palabra. Nosotros colocamos un grano de arena inicial para esta tarea.

Digite el listado siguiente:

```
10 GRAPHICS 20
20 DIM JX(15),JY(15)
30 FOR I=5 TO 15
40 READ X,Y
50 JX(I)=X:JY(I)=Y
60 NEXT I
70 DATA 1,1,1,-1,1,0,0,0,-1,1,-1,-1,-1
,0,0,0,0,1,0,-1,0,0
80 FOR J=1 TO 7
85 PX=J*10:PY=20
90 COLOR 1
92 RESTORE 99:FOR K=1 TO 10:READ S
93 PLOT PX,PY
94 GOSUB 170:A=1^1^1
96 NEXT K:NEXT J
98 END
99 DATA 6,6,7,13,13,13,13,13,13,13
100 PX=40:PY=20
110 COLOR 1
120 PLOT PX,PY
130 S=STICK(0)
140 IF S=15 THEN 110
170 PX=PX+JX(S)
180 PY=PY+JY(S)
200 IF PX>79 THEN PX=0
210 IF PY>39 THEN PY=0
220 IF PY<0 THEN PY=39
230 IF PX<0 THEN PX=79
240 RETURN
```

Ejecute el programa y observará como en la pantalla se repite siete veces el dibujo del número uno.

Con esta primera etapa se muestra al interlocutor (peque) como deben realizarse los trazos para dibujar el número uno.

Pero nos falta la parte más importante. ¿Cómo interactúa este pequeño con el ATARI?

Agregue las siguientes líneas al programa inicial:

```
98 GOTO 300
300 GRAPHICS 20:RESTORE 99:PX=40:PY=20
310 COLOR 1:PLOT PX,PY
320 FOR K=1 TO 10:READ D
330 S=STICK(0):A=1^1^1^1
335 IF S=15 THEN 330
340 IF S=D THEN 360
350 FOR Z=1 TO 255:SOUND 0,Z,10,6:NEXT
 Z:SOUND 0,0,0,0
355 GOTO 330
360 A=1^1^1^1
365 GOSUB 170:PLOT PX,PY
370 NEXT K
```

Ejecútelo y observe las posibilidades de este programa en otras actividades.

MUNDOATARI, en sus columnas PEQUES y CLASES, tiene abierta la puerta para recibir contribuciones de profesores de los diferentes niveles, que hayan sido tocados por ATARI en la senda de educación computacional. Lean al respecto nuestro aviso de colaboraciones.

Hasta el próximo mes. ●

MINIMO COMUN MULTIPLO
vuelve a la carga

En relación al programa para identificar el Mínimo Común Múltiplo para 2 listas, publicado en MUNDOATARI No. 2, página 31 y cuyo autor fue don Daniel Hasler, transcribimos un par de comentarios hechos telefónicamente por un socio del Club ATARI, que pidió no ser mencionado.

MUNDOATARI accedió a esa petición, atendiendo a que se trata de comentarios constructivos y bien intencionados.

Los cambios sugeridos son:

1. El algoritmo ubicado en la línea 110 que busca la identidad de los múltiplos mediante la división entre ellos hasta que el resultado sea 1, puede ser reemplazado por el operador "=".

2. La búsqueda del Mínimo Común Múltiplo puede quedar reducida al primer múltiplo que aparezca y así hacer honor al título.

Los dos cambios se efectúan en la misma línea:

110 IF FACT1(X) = FACT2(Z) THEN PRINT "Comun Multiplo = " ; FACT 1(X): END

La ventaja de estos cambios es la mayor rapidez de respuesta: se gana casi un 50o/o de tiempo.

MUNDOATARI agradece ambos aportes y los comunica a los lectore para su consideración.

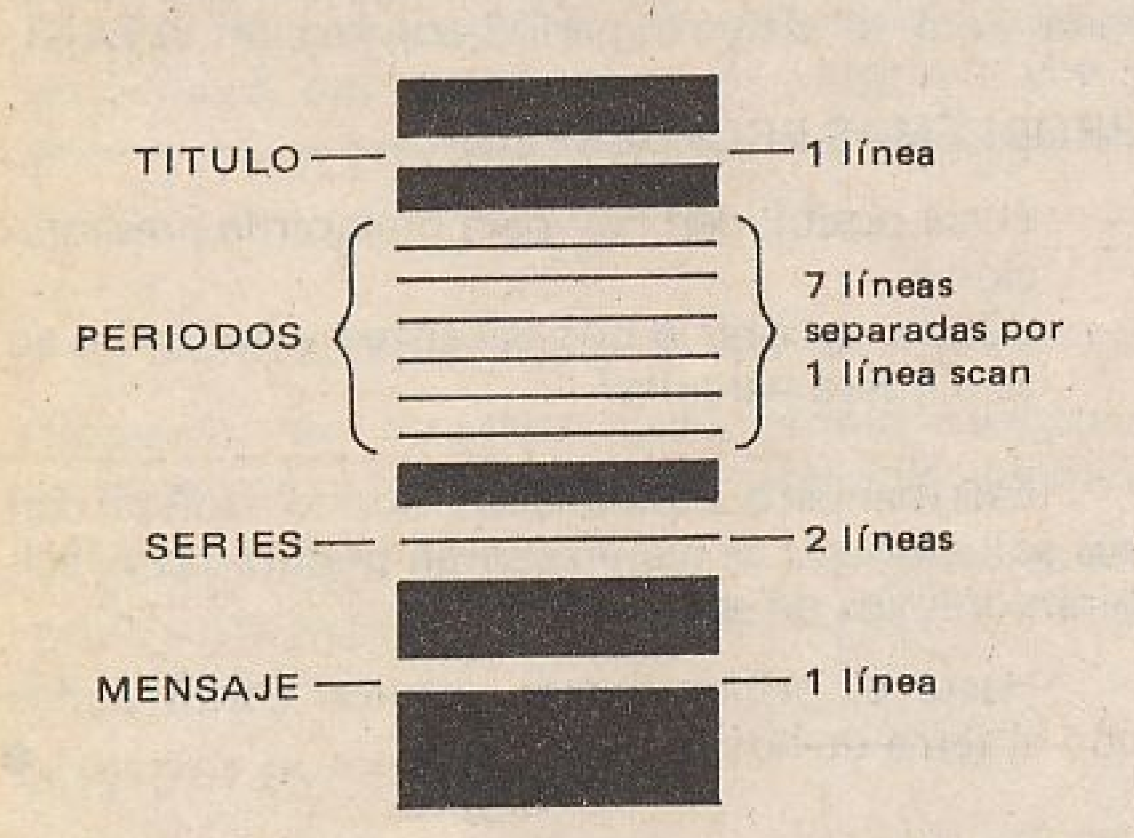
Clases con ATARI

Nuestra columna se dirige esta vez a los mayores de Enseñanza Media y nuestro tema será el **Sistema Periódico de Elementos.**

Este interesante tema lo desarrollaremos juntos a nivel de aplicaciones.

La pantalla es el periférico que presentará el aspecto del sistema periódico.

En el esquema siguiente observamos los requerimientos de pantalla para desplegar la tabla:



- Una línea superior para el título,
- Un espacio separador,
- Siete líneas horizontales que corresponden a los períodos. Entre ellas debe existir una línea scan separadora.

NOTA: El término **período**, como concepto químico, es un grupo de elementos que pertenecen a una misma línea horizontal de la tabla.

- Un espacio separador,
- Dos líneas horizontales para las series de elementos,
- Dos espacios separadores,
- Una línea inferior de comunicación con el usuario.

¿Cómo crear este formato de pantalla?

Los modos gráficos normales no resuelven este problema. La técnica de Lista de Despliegue, analizada en detalle en MUNDOATARI Nos. 0, 1, 2 y 3, nos da la información adecuada.

Digite el siguiente listado:

```
1000 DATA 112,112,112,66,0,144,112,66,
0,145,0,66,0,146,0,66,0,147,0,66,0,148
,0,66,0,149,0
1010 DATA 66,0,150,0,66,0,151,112,66,0
,152,0,66,0,153
1020 DATA 112,112,66,0,154
1030 DATA 65,0,6,-1
```

Estas líneas de DATA proporcionan la información al ANTIC sobre las instrucciones de Lista de Despliegue para el formato requerido.

Interprete estos datos como una aplicación de los conocimientos adquiridos en los números anteriores y escríbanos comentándolos. Hay premios para las respuestas correctas.

Estos datos se incorporarán en la página seis de la memoria (inicio en la localización 1536).

PARTICIPANDO

MUNDOATARI publica este mes la participación de Gonzalo Soto, atariano de apenas 7 años de edad, quien envía su última creación para mostrar los 256 colores en la pantalla del televisor.

```
1 REM PARTICIPANDO PEQUE
2 REM GONZALO SOTO SILVA
3 REM SAN FRANCISCO 9060
5 REM PUDAHUEL
6 REM SANTIAGO
7 REM ******************************
10 GRAPHICS 0
20 PRINT "COLORES DE LOS POKES"
30 FOR F=0 TO 255:POKE 710,F:POKE 712,
F
40 POSITION 4,10:PRINT "POKE NUMERO ";
F
50 FOR R=1 TO 500:NEXT R
60 POSITION 10,20:PRINT "G.A.S.S."
70 NEXT F
80 END
145 GOSUB 2000
```

En premio por la participación puedes elegir 2 programas de la lista publicada en MUNDOATARI No. 2, páginas 38-39, que serán enviados a tu domicilio a vuelta de correo.

ERRATA

En MUNDOATARI No. 1, página 14, se omitió la línea 35 del programa demostrativo en la actividad 2. En consecuencia debe quedar el listado como sigue:

```
10 REM PROGRAMA # 1
20 FOR J=0 TO 255 STEP 16
30 POKE 710,J
35 FOR K=1 TO 200:NEXT K
40 NEXT J
50 END
```

Se le envió un estímulo al lector que proporcionó telefónicamente la advertencia y solución.
Muchas gracias ... y continúe vigilante.

El siguiente listado efectúa dicho traspaso:

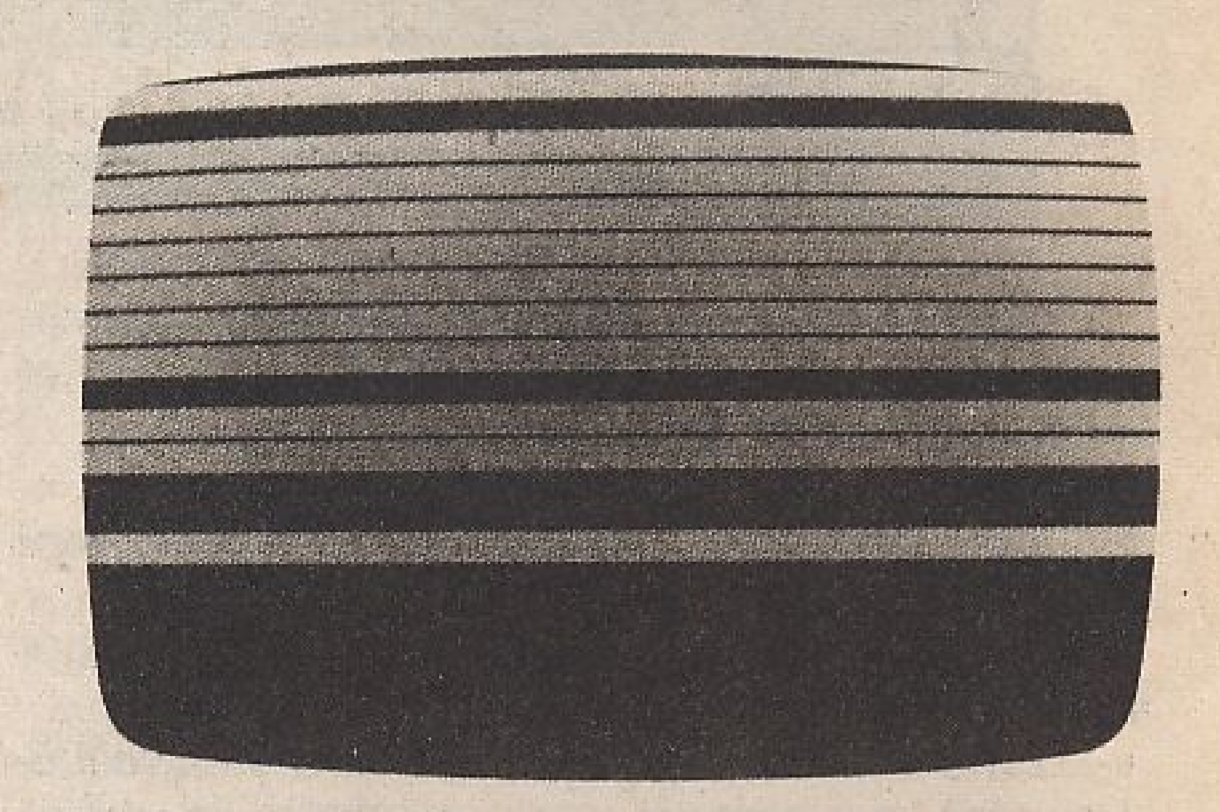
```
100 RESTORE 1000:LOC=1536:INC=0
110 READ D:IF D=-1 THEN 160
120 POKE LOC+INC,D
130 INC=INC+1:GOTO 110
```

Ejecute el programa y observe la pantalla. ¿Sucede algún cambio? ¿Qué es lo que falta?

Digite la línea siguiente:

```
160 POKE 560,0:POKE 561,6
```

Ejecute el programa nuevamente y observe:



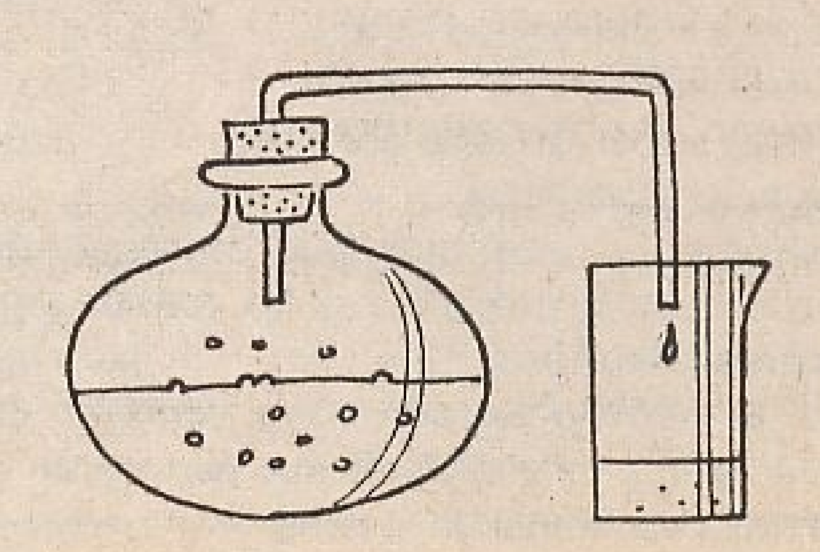
¿Qué le parece? He aquí la presentación definida para el sistema periódico, versión ATARI.

PROBLEMAS PROPUESTOS:

- ¿Qué posibilidad hay para mejorar la presentación?
- ¿Cómo escribir datos del sistema periódico en esta nueva pantalla?

Escríbanos a casilla 458, Correo Ñuñoa con sus soluciones. Las mejores serán publicadas y recibirán premios de estímulo.

Hasta el próximo mes, cuando continuemos con el tema de hoy. ●



Proyectos

Cuarta parte



Estimados lectores, este mes haremos una breve introducción para explicar en esta columna el motivo de las interrupciones en sus participaciones a la columna. Las revistas a suscriptores llegan aproximadamente 10 ó 15 días antes que a los kioskos, lo que implica una fecha de cierre de la edición de casi un mes a la fecha de salida.

Por esta razón las actividades para nuestros lectores se publicarán en el siguiente número de septiembre, que aparecerá en los kioskos el día 3.

Ahora vamos al tema central de la columna:

Crearemos un programa que permita manipular un banco de datos de notas del listado de alumnos.

En este número cambiaremos la estrategia de números anteriores, por no contar con el soporte de otras columnas.

ACTIVIDAD:

— Crear las líneas de programa que permitan:
- abrir el archivo PRUEBA.DAT
- ingresar 20 notas al archivo.

Esta actividad es una aplicación de contenidos de este número para la columna Dominando la 1050.

Repasar previamente esos conceptos le puede ayudar a entender mejor esta columna o a comparar los modelos en cada caso.

```
5 DIM T$(3)
10 OPEN #1,8,0,"D:PRUEBA.DAT"
20 FOR J=1 TO 20
30 PRINT "DIGITE NOTA  DE ";J
40 INPUT T$
50 PRINT #1;T$
60 NEXT J
70 CLOSE #1
```

Este listado tiene la respuesta para la actividad. Digítelo cuidadosamente.

Ejecútelo e ingrese 20 notas de asignatura. Anote paralelamente las notas ingresadas para verificarlas en la actividad siguiente.

La foto ilustra el momento de la ejecución:

```
?6
DIGITE NOTA  DE 11
?7
DIGITE NOTA  DE 12
?5
DIGITE NOTA  DE 13
?5
DIGITE NOTA  DE 14
?5
DIGITE NOTA  DE 15
?5
DIGITE NOTA  DE 16
?5
DIGITE NOTA  DE 17
?5
DIGITE NOTA  DE 18
?5
DIGITE NOTA  DE 19
?5
DIGITE NOTA  DE 20
?5
```

Salve estas líneas de programa con el nombre: PROYE1.AGO

ACTIVIDAD:

— Crear las líneas de programa que permitan:
  - leer las 20 notas almacenadas en el diskette
  - comprobar si las notas corresponden a las digitadas anteriormente:

```
5 DIM T$(3)
100 CLOSE #1:OPEN #1,4,0,"D:PRUEBA.DAT
"
110 TRAP 180:GRAPHICS 0
120 FOR J=1 TO 20
130 INPUT #1;T$
140 PRINT "LA  NOTA  DE ";J;" ES ";T$
150 NEXT J
160 ? " PRESIONE RETURN Y CONTINUA ":I
NPUT T$
165 GOTO 120
170 CLOSE #1
180 END
```

Este segundo listado tiene la solución.

¿Se preocupó de salvar primero el programa PROYE1.AGO?. Si es así puede continuar. En caso contrario salve el programa.

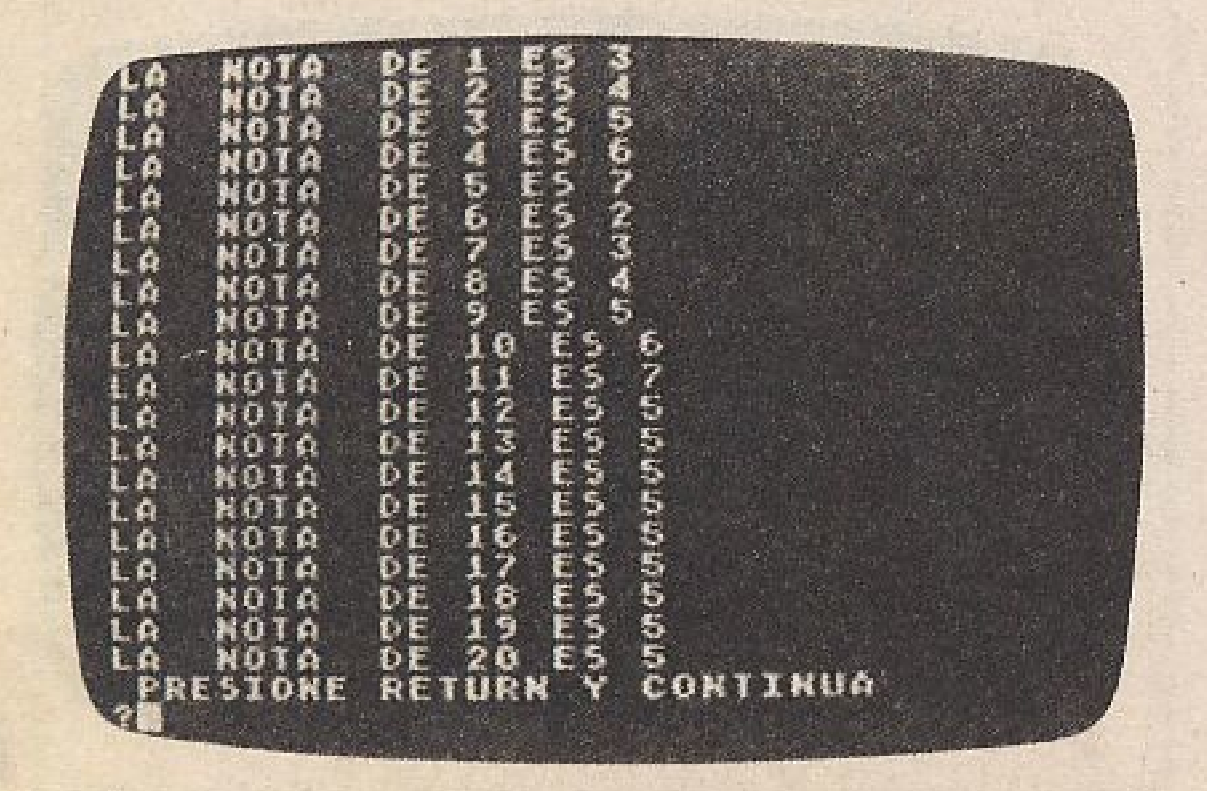
Digite NEW y su memoria ha borrado el programa existente. Digite LIST y RETURN para comprobarlo.

A continuación digite el programa y ejecute. Compruebe las notas ingresadas.

Salve este programa con el comando LIST y el nombre PROYE2.AGO

¿Qué importancia tiene utilizar la instrucción LIST para grabarlo?

La foto ilustra un momento del rescate de notas.



ACTIVIDAD:

— Crear las líneas de programa que permitan:
  - agregar una nueva lista de notas a las existentes

Borre el programa residente en memoria según el procedimiento anterior.

Este listado tiene la solución:

```
5 DIM T$(3)
200 OPEN #1,9,0,"D:PRUEBA.DAT"
210 GRAPHICS 0
220 FOR J=1 TO 20
230 PRINT "DIGITE NOTA  DE ";J
250 INPUT T$
255 PRINT #1;T$
260 NEXT J
270 CLOSE #1
```

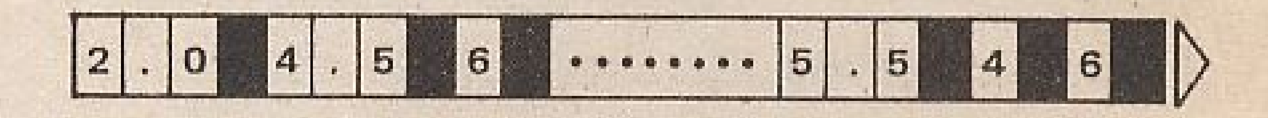
Digítelo cuidadosamente. Ejecute e ingrese otras notas.

Anótelas para verificarlas en la actividad siguiente.

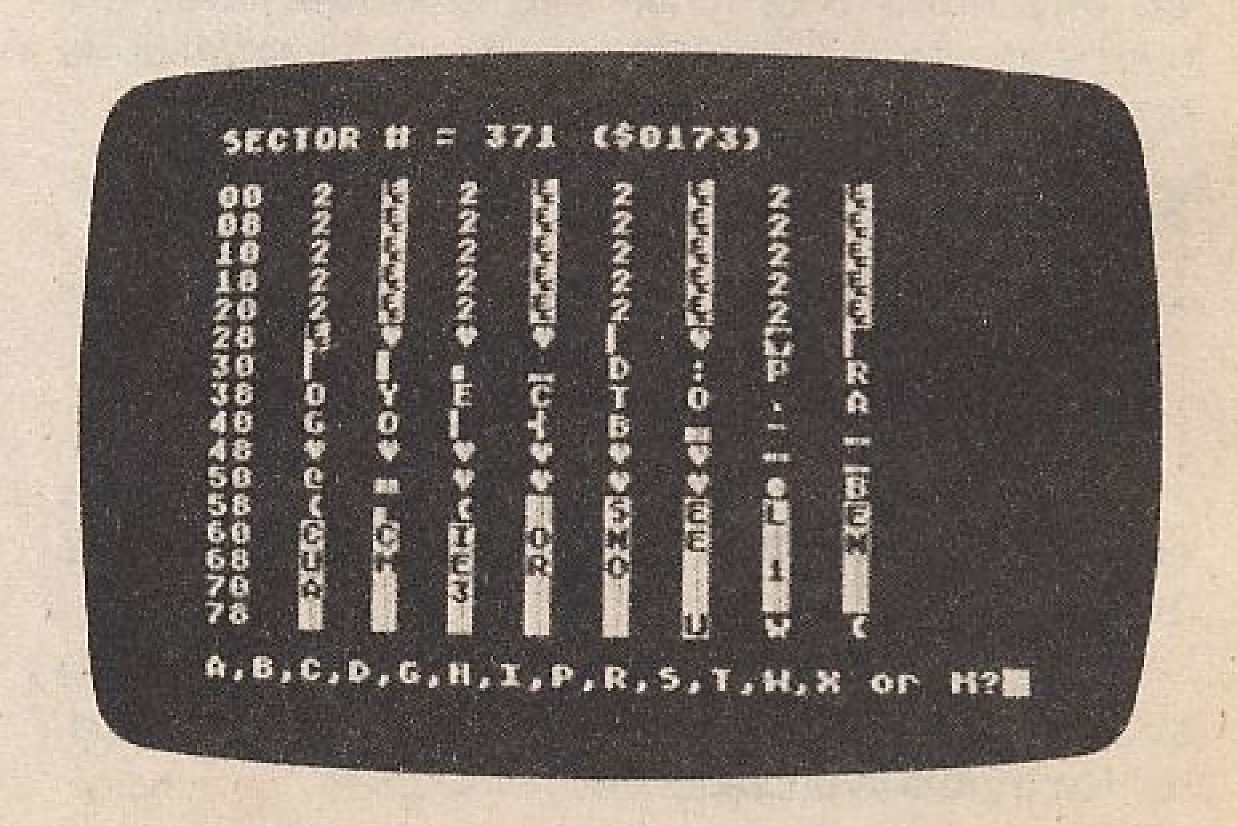
Salve este programa como PROYE3.AGO con la instrucción LIST.

En el diskette deberían existir dos listas de notas.

El esquema siguiente ilustra la ubicación de los archivos:



La fotografía ilustra la disposición en sectores del diskette.



ACTIVIDAD:

- **¿Cuál de los tres programas digitados permite rescatar las notas desde el archivo en diskette hacia la memoria y presentarlo en pantalla?**

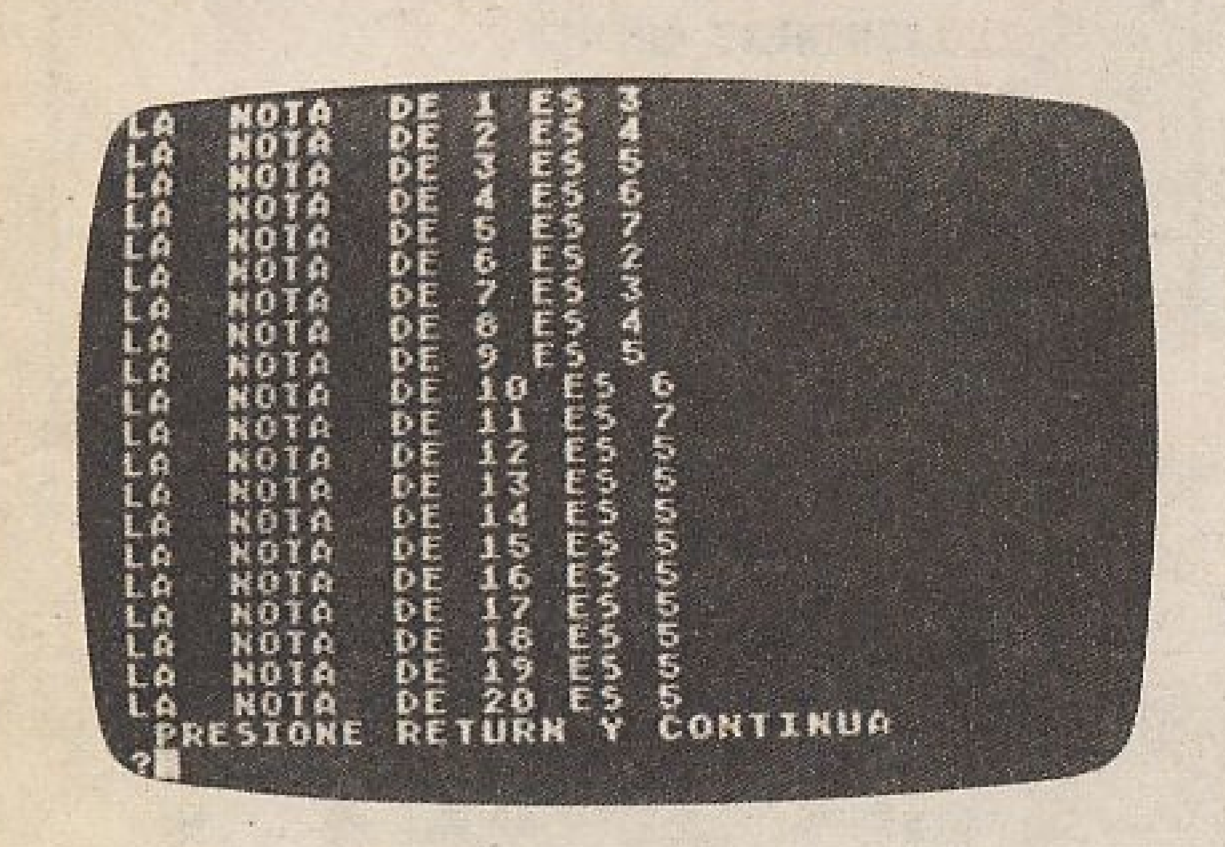
Brillante respuesta si se decidió por el segundo listado PROYE2. AGO

Antes de continuar borre el programa en memoria.

Cargue el programa PROYE2.AGO, con la instrucción ENTER.

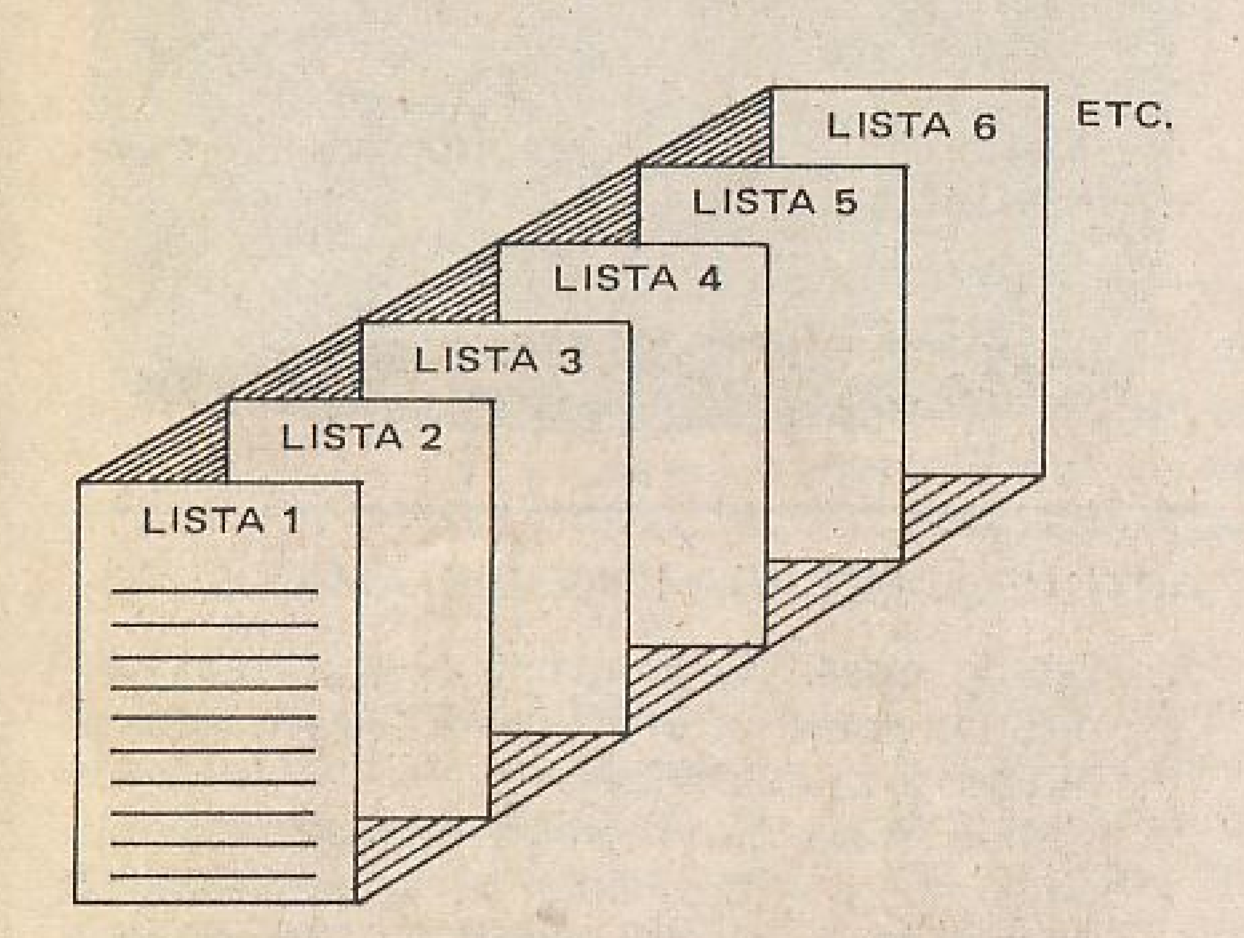
Ejecute y verifique la presencia en el archivo de dos listas de notas.

La foto ilustra un ejemplo de ejecución:



En consecuencia, parte de la tarea está cumplida con estos tres módulos. El esquema ilustra su funcionalidad:





Este archivo creado es un **archivo secuencial.**

¿Cómo interactuar con estos módulos en un mismo programa?

ACTIVIDAD:

- **Cargue los 3 programas PROYE1, PROYE2 y PROYE3 usando las instrucciones adecuadas**

Para poder interconectar estos módulos se necesita un MENU.

ACTIVIDAD:

- **El siguiente listado proporciona un simple modo de direccionamiento:**

```
1000 GRAPHICS 0
1010 POSITION 10,5:? "   M E N U"
1020 POSITION 5,8:? " 1  INICIA ARCHIV
O"
1030 POSITION 5,10:? " 2  AGREGA NUEVA
 LISTA NOTAS"
1040 POSITION 5,12:? " 3  VE LISTAS NO
TAS"
1050 POSITION 0,15:? " SELECCIONE NUME
RO 1 A 3          "
1060 INPUT N
1070 ON N GOTO 10,200,100
```

Agregue las siguientes líneas:

```
9 GOTO 1000
80 GOTO 9
180 GOTO 9
280 GOTO 9
```

Ejecute el programa y realice las siguientes pruebas:

- Modifique las líneas 10, 100 y 200, cambiando el nombre de archivo por PRUEB2.DAT
- Seleccione las opciones para que compruebe el movimiento del puntero.

La opción 1 borra todos los registros existentes, pues mueve el puntero a la posición inicial (8).

La opción 2 agrega registros a los existentes y permite **conservar los anteriores** (9).

La opción 3 permite leer todos los registros desde la posición inicial (4).

¿Cómo se ingresaron estas notas?

Las posibilidades de ingreso de información son:

- como variable numérica
- como variable alfanumérica (string).

La dimensión de la variable T$ nos indica que el ingreso es a variable alfanumérica.

ACTIVIDAD:

— Ahora nos preocupa diseñar la estructura que permita almacenar los valores (notas) y realizar los cálculos.

El listado siguiente proporciona la respuesta:

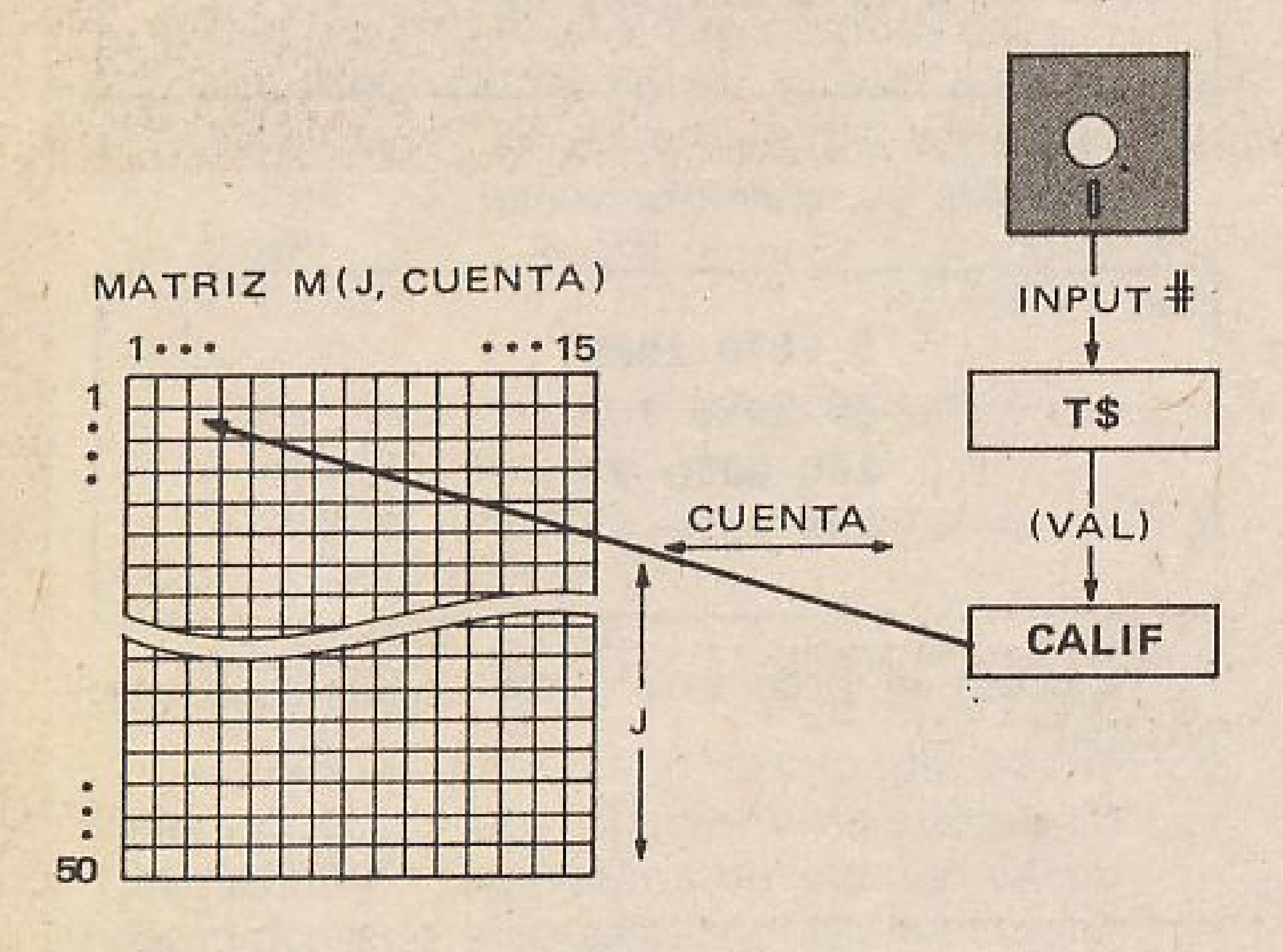
```
2000 REM
2010 CALIF=VAL(T$)
2020 M(J,CUENTA)=CALIF
2030 RETURN
```

Agregue estas líneas y ejecute el programa.

```
7 DIM M(50,15)
145 GOSUB 2000
```

¿Existen cambios en la ejecución?. No, sólo se almacenan los valores de las notas como variables numéricas con doble índice en la matriz M.

La funcionalidad de las líneas se representan en el siguiente esquema:



Esta es una subrutina que se accesa por cada valor ingresado desde el diskette mediante la instrucción INPUT #1; T$.

La variable alfanumérica T$ es auxiliar en el tránsito hacia la celda de la matriz.

Detengámonos en la línea:

```
2020 M(J,CUENTA)=CALIF
```

Esta contiene el movimiento direccional de los valores ingresados temporalmente en T$ hacia las diferentes celdas de la matriz M.

La variable J es la contadora del ciclo de ingreso desde el diskette y se desplaza verticalmente a lo largo de una columna.

La variable CUENTA se desplaza de una columna a otra en una misma línea.

NOTA: Es importante que al usar una matriz se inicialice previamente, de lo contrario pueden acontecer situaciones trágicas.

Las líneas 2100-2150 contienen una subrutina para inicializar la matriz M.

```
2100 FOR J=0 TO 50
2110 FOR CUENTA=0 TO 15
2120 M(J,CUENTA)=0
2130 NEXT CUENTA
2140 NEXT J
2150 RETURN
```

La siguiente línea llama esa rutina:

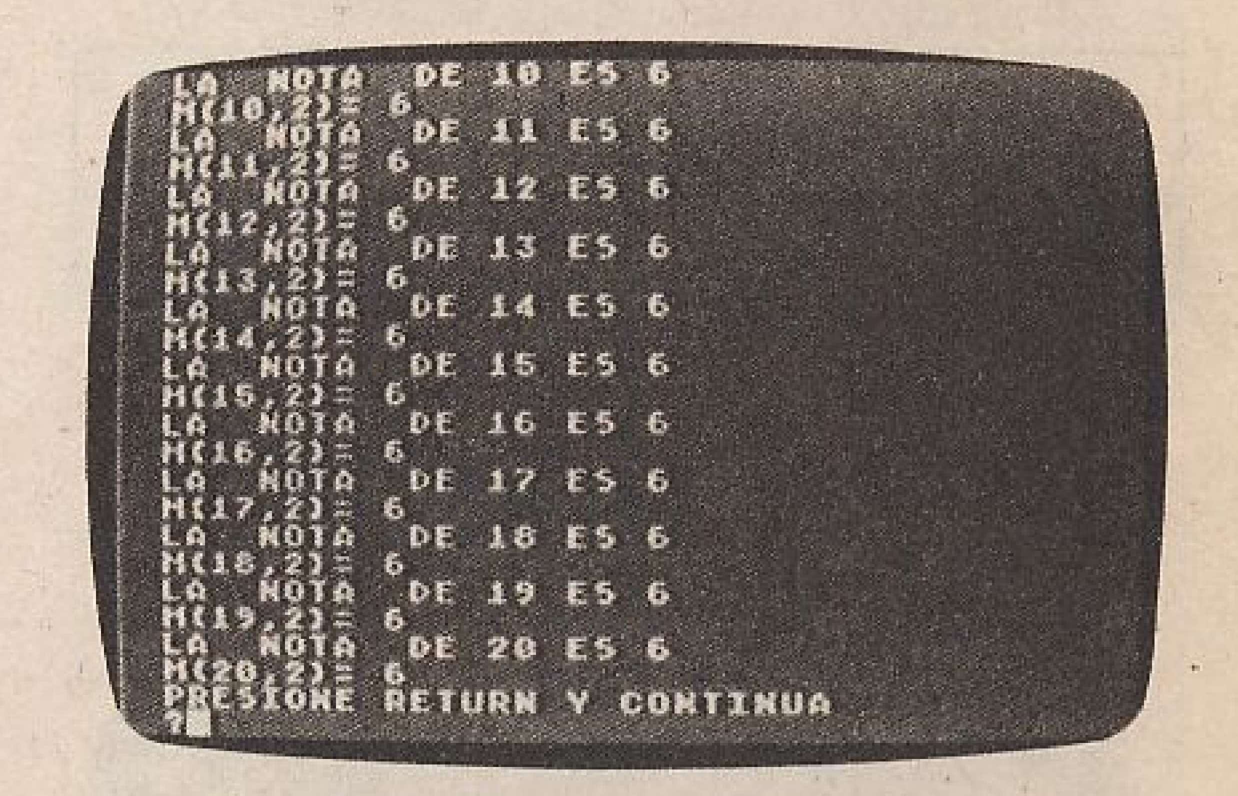
GOSUB 2100

ACTIVIDAD:

— Diseñe las líneas que permitan visualizar el contenido de una columna de notas.

```
150 PRINT "M(";J;",";CUENTA;")= ";M(J,
CUENTA)
```

Al ejecutar observará en pantalla como lo ilustra la foto:



ACTIVIDADES PARA USTEDES, AMIGOS

1. Diseñe líneas de programa que permitan sumar las notas de una línea y sacar el promedio de las calificaciones.
2. Diseñe líneas de programa que permitan obtener el valor promedio de la prueba. Estadística: suma de las notas de una columna dividido por el número de líneas.





Music Studio

Este programa de alta sofisticación permite comprobar las excelentes capacidades de la interfase MIDI de la línea de computadores ATARI ST

Una magnífica herramienta para usuarios con habilidades en formas de expresión musical, caracterizada por una forma amistosa de comunicación, que en poco tiempo le permitirá conocerlo en su mayor extensión.

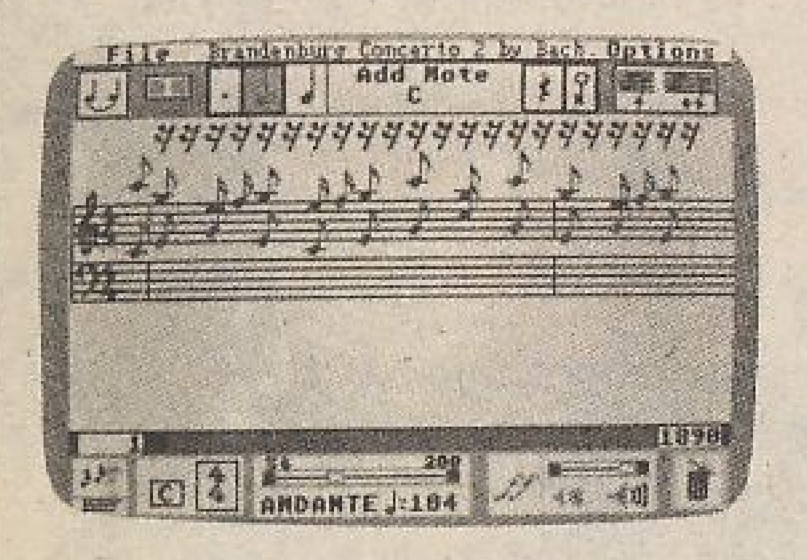
Puede utilizarse con un sintetizador adicional o solamente con el computador. La diferencia es que en el primer caso podrá tener acceso a un mayor número de canales.

Para ejecutarlo debe activar con el mouse el programa en floppy disk AUDIO.PRG, después del cual aparece una foto característica de presentación en la pantalla y emergen simultáneamente exitantes sonidos de su computador.

Después de la pantalla de inicio y del intermedio musical que se presenta a continuación:



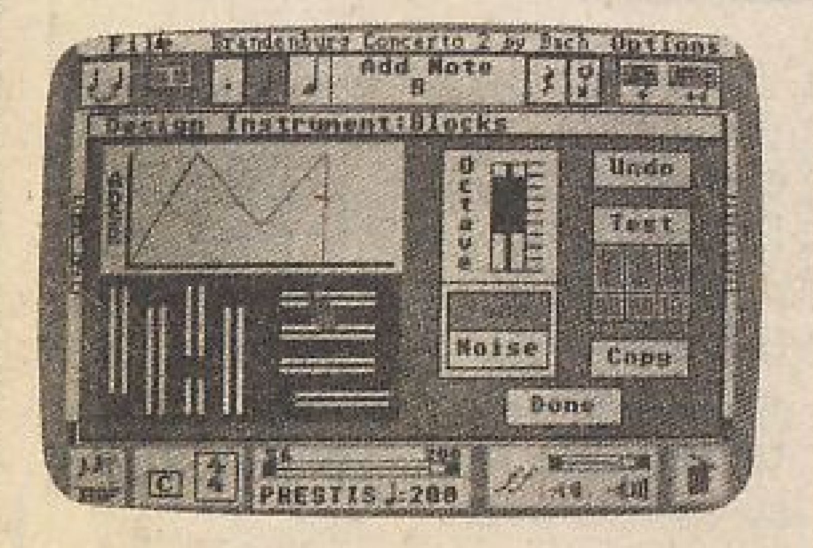
se encuentra ante un pentagrama como el que ilustra la foto siguiente:



Existen una serie de comandos superiores que le permiten poner en acción su capacidad creadora musical.

Para efectos de hacer una demostración puede cargar los diferentes programas que se encuentran en el archivo del diskette.

Pida una demostración de este programa en los Centros ATARI o en los distribuidores de la línea ST y apreciará las bondades del programa y la facilidad con que puede introducir en su casa una orquesta de quince instrumentos, dirigida por usted mismo.



MUSICAL INSTRUMENT DIGITAL INTERFASE:

MIDI

Hasta hace algunos años, la música era lo que estaba más alejado de la computación, ya que los sonidos distaban mucho de ser compatibles con la electrónica digital.

Hace poco tiempo se creó la interfase **MIDI**, Musical Instrument Digital Interfase, la cual permite a los instrumentos musicales conectarse a los computadores y así aumentar la gama de sonidos y generar prácticamente una orquesta con un sólo instrumento. Pero esta interfase es aún muy cara y tenía que adosársele al computador como un periférico más.

ATARI solucionó ese problema, incorporando a su línea de computadores ST la interfase MIDI dentro del computador y disponiendo del software necesario para manejarla.

Se debe tener cuidado con los instrumentos que se conecta al computador, ya que no todos traen la interfase MIDI incorporada. Normalmente se conectan todos los instrumentos de tipo electrónico (sintetizadores, teclados, guitarra eléctrica, etc.). Nosotros hemos tenido muy buenos resultados con el teclado del sintetizador y ocupando el programa ST Music Studio.

Conecte el teclado al computador, cargue el programa y haga la siguiente prueba: presione cualquier nota en el teclado y ésta aparecerá en el pentagrama que está en la pantalla. El programa le permitirá "escribir" su música para luego listar todas las notas por impresora, hacer algún arreglo musical en el cual puede tener hasta 16 instrumentos distintos (1 por canal) tocando al mismo tiempo y componer sus propios temas.

Usted no debe preocuparse de la métrica del tema, el computador lo hace por usted. Además tiene todos los tipos de duración de nota existentes y por supuesto sus respectivos silencios.

Otra de las características de la MIDI es que usted puede generar su propio instrumento. Esto se hace deformando la onda musical del piano, o del instrumento que tenga el teclado que usted usa normalmente.

Si usted crea por ejemplo una canción y le hace un arreglo de cuatro instrumentos (batería, bajo, piano y guitarra), puede grabarlo en diskette y luego ejecutarlo completo sin que usted toque una sola tecla, o bien, al momento de la ejecución usted puede agregar un quinto instrumento tocándolo usted mismo al momento de la ejecución sin perjuicio para los restantes.

Existen varios otros programas que le permiten usar las MIDI, enfocados a distintas áreas de la música, por ejemplo el META TRACK le permite "grabar" por pistas los distintos sonidos del teclado y luego reproducirlos por partes o como un todo.

Esperamos que estas líneas les den una clara idea del uso de la interfase MIDI, de modo que quien use esta opción de la computación-musical pueda llegar a ser un Ludwig van Beethoven o un Wolfgang Mozart de la era electrónica.

Juegos para ST

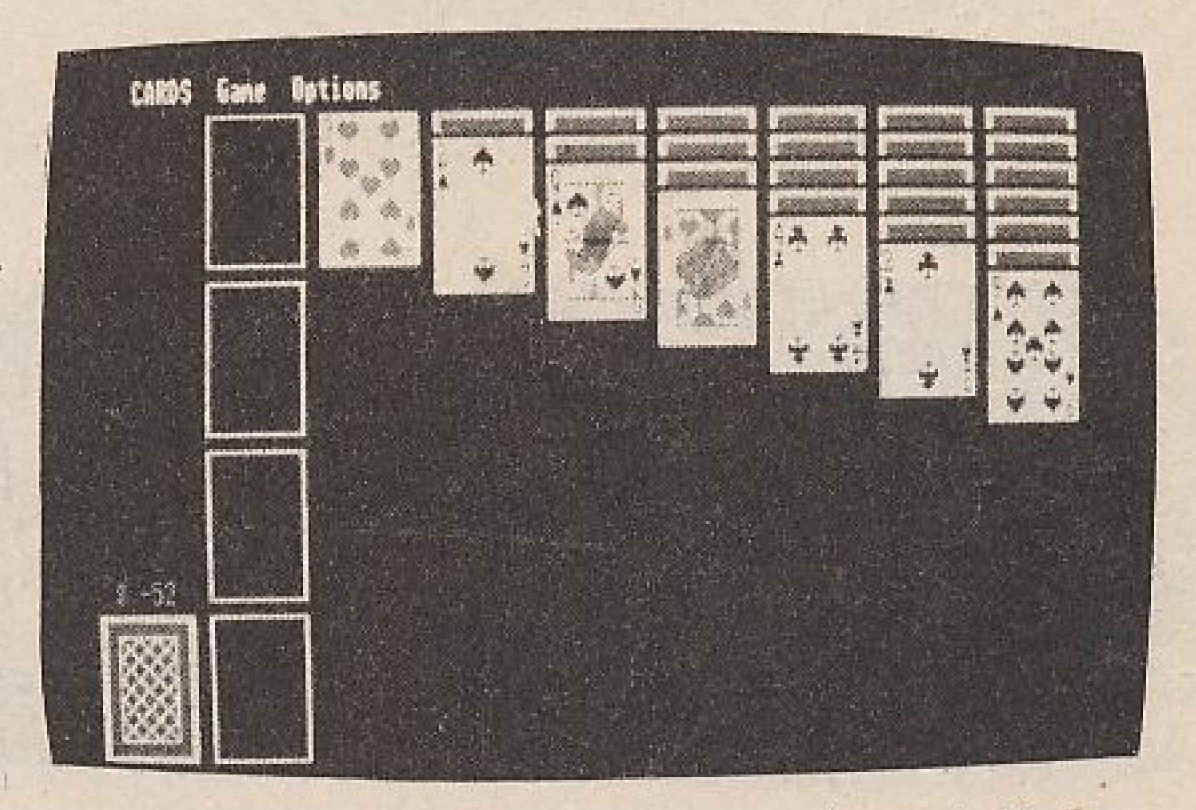
Presentamos a continuación una serie de juegos que usted dispone para la línea ST, con una capacidad gráfica extraordinaria.



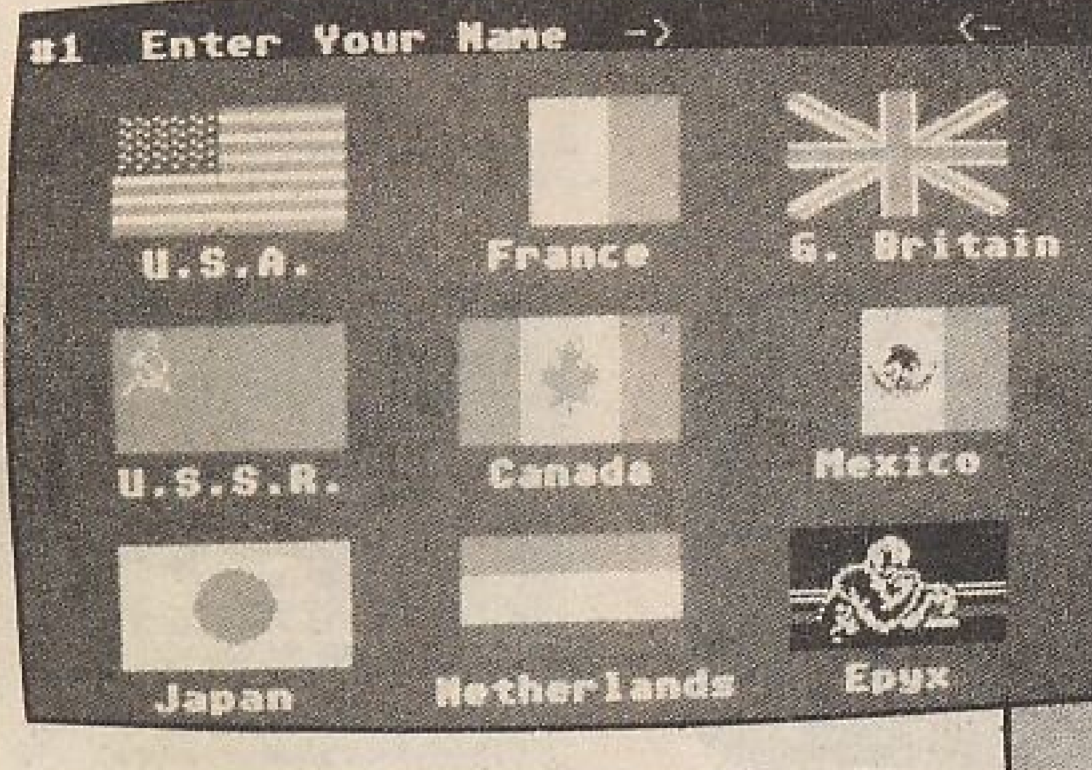
Super Cycle

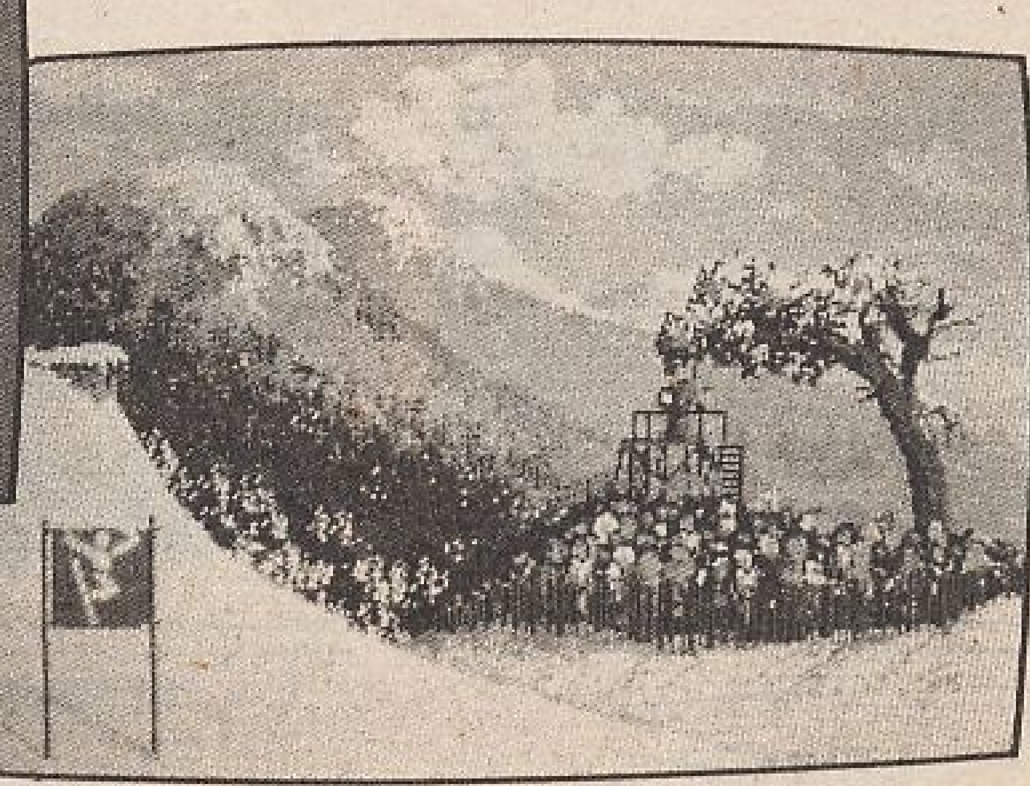


Rogue



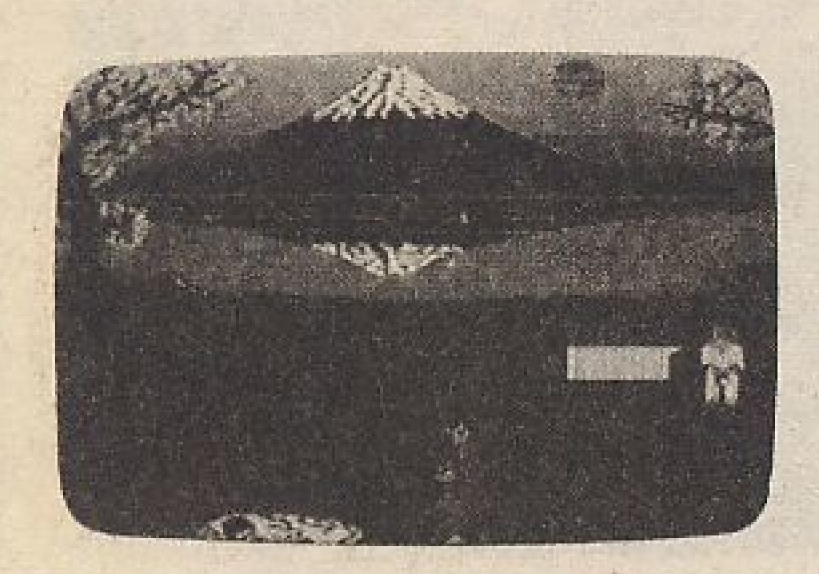
Juego de Naipes





Winter Games

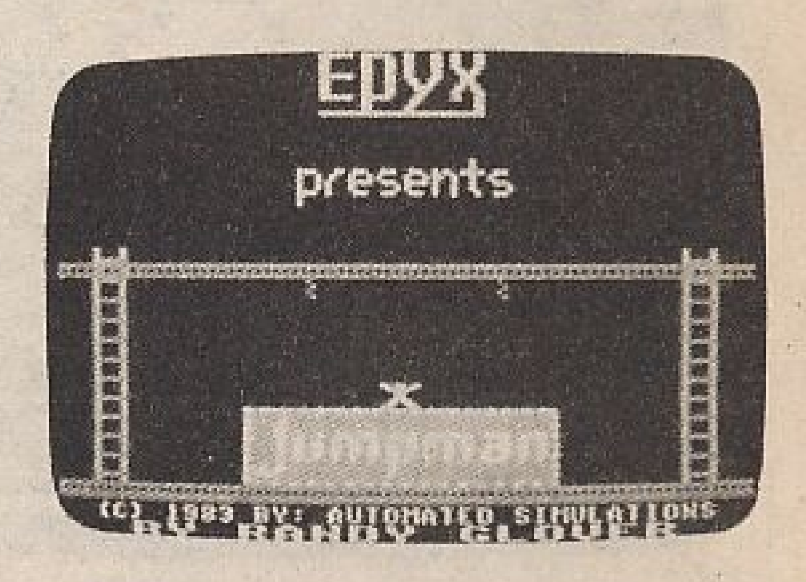




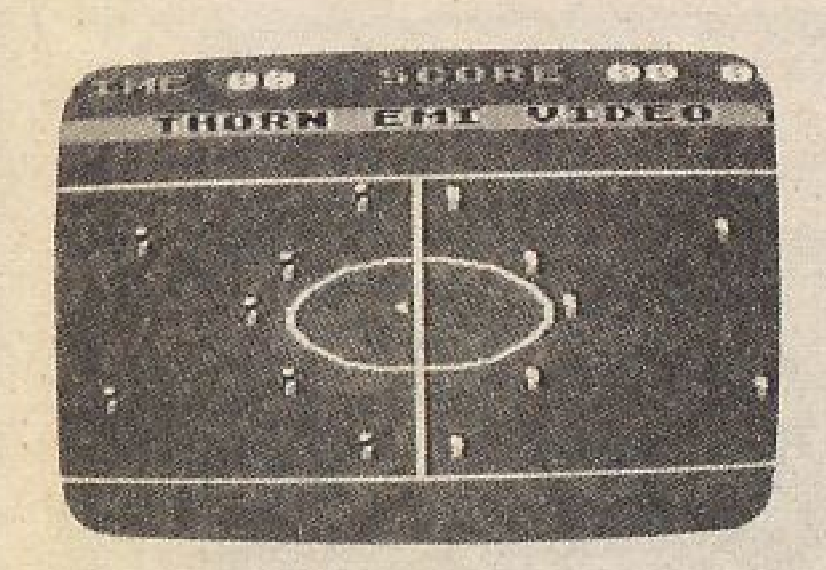
CHAMPIONSHIP KARATE
– 100 puntos



SILENT SERVICE
– 60 puntos



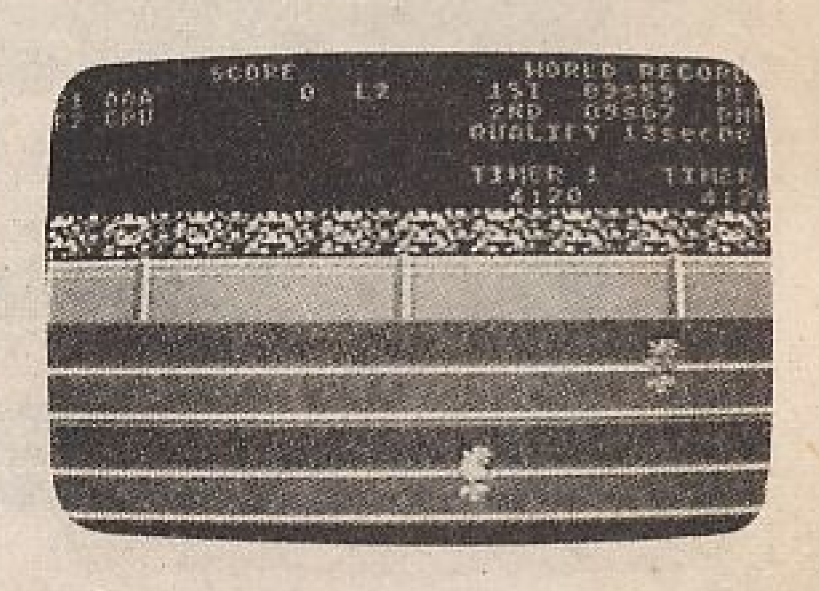
JUMPMAN
– 50 puntos



FUTBOL
– 20 puntos



KENNEDY APROACH
– 15 puntos



TRACK & FIELD
– 15 puntos

VIDEOMANIA

AMIGO LECTOR:

CON LAS CARTAS QUE ENVIES
A ESTA COLUMNA
PUEDES PARTICIPAR
EN ESTE RANKING
Y GANAR PREMIOS.

¡PARTICIPA!

PREMIOS DEL MES

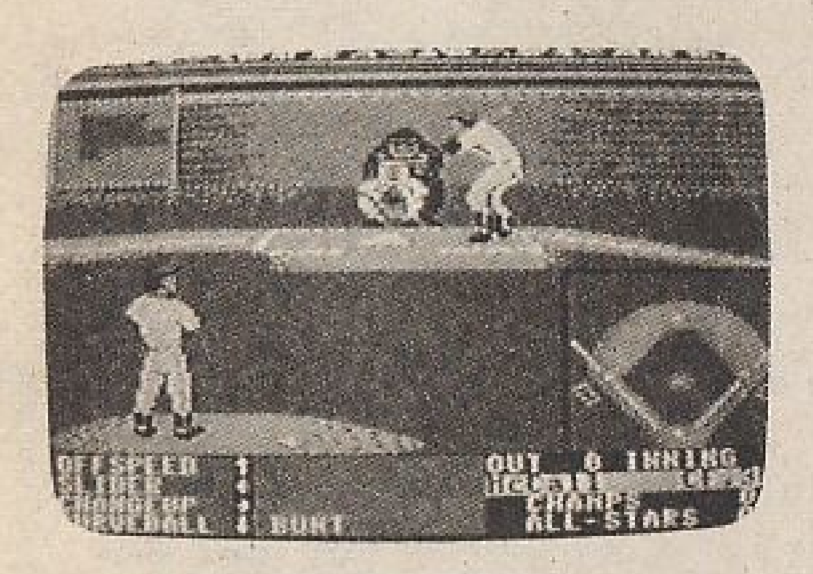
Con las cartas recibidas hasta el 5 de julio se ha efectuado la primera selección, en la cual resultaron premiados los siguientes videomaníacos:

1. *Graciela Hüper Gómez, de Santiago*
   No. socio: 82-64-4000-0
2. *Andrés González, de San Bernardo*
   No. socio: 60-44-6001-3
3. *Alvaro Sandoval, de Temuco*
4. *Christián Rivera C., de Coquimbo*
   No. socio: 40-10-8001-9

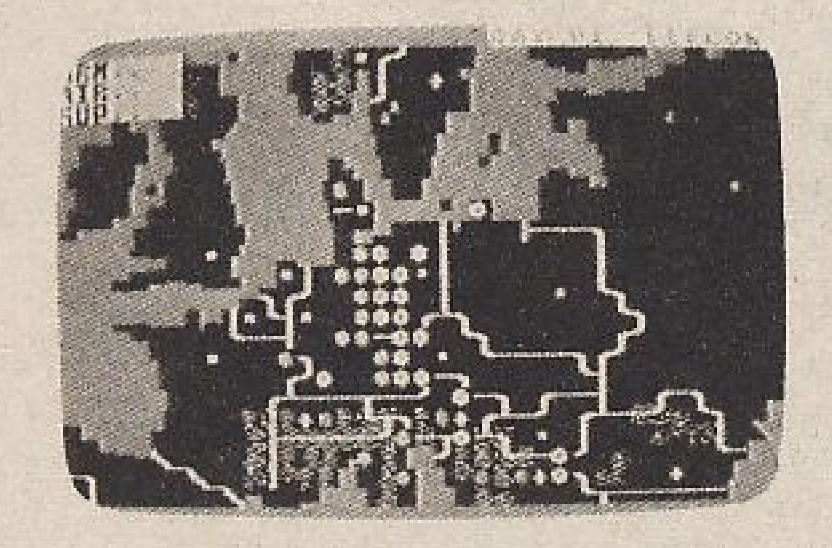
Los premios de estímulo han sido enviados a sus respectivos domicilios.

NOTA A LOS PARTICIPANTES:

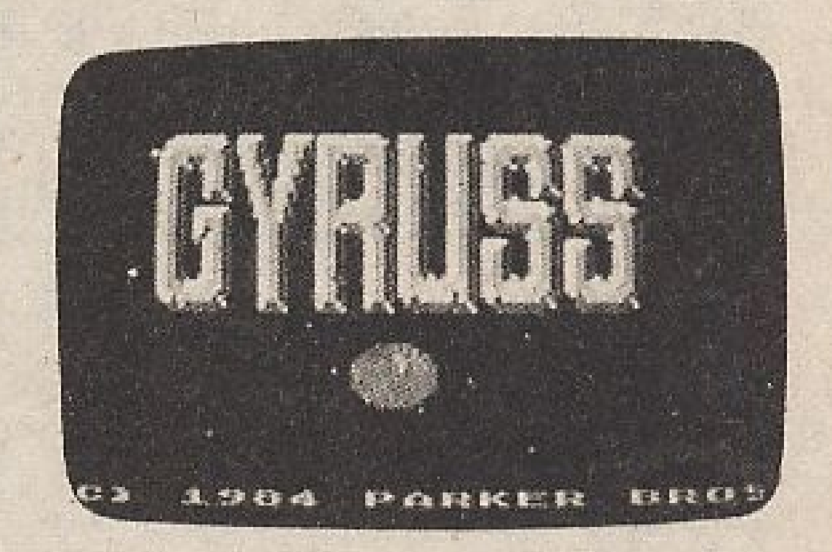
Para que los premios sean apropiados a tu configuración, indica el tipo de periférico que dispones (casetera o diskettera)



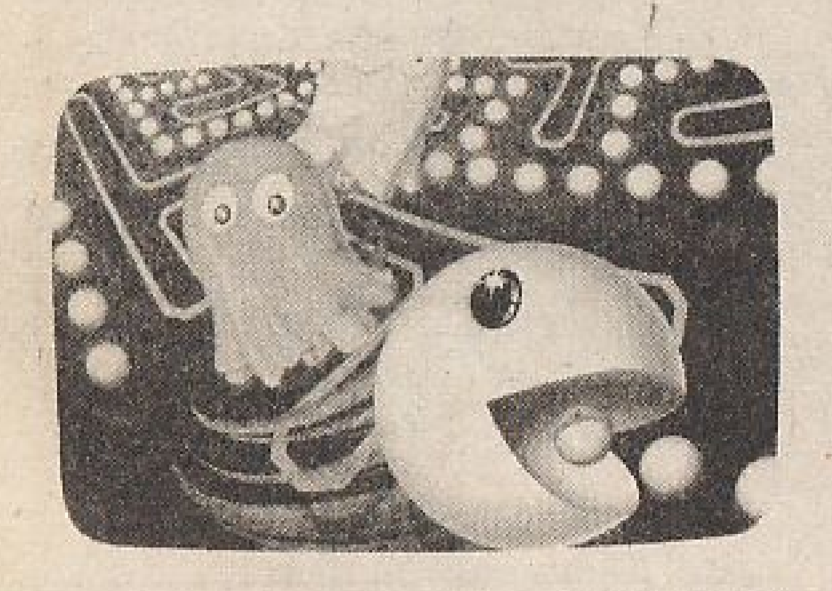
BEISBOL
— 30 puntos



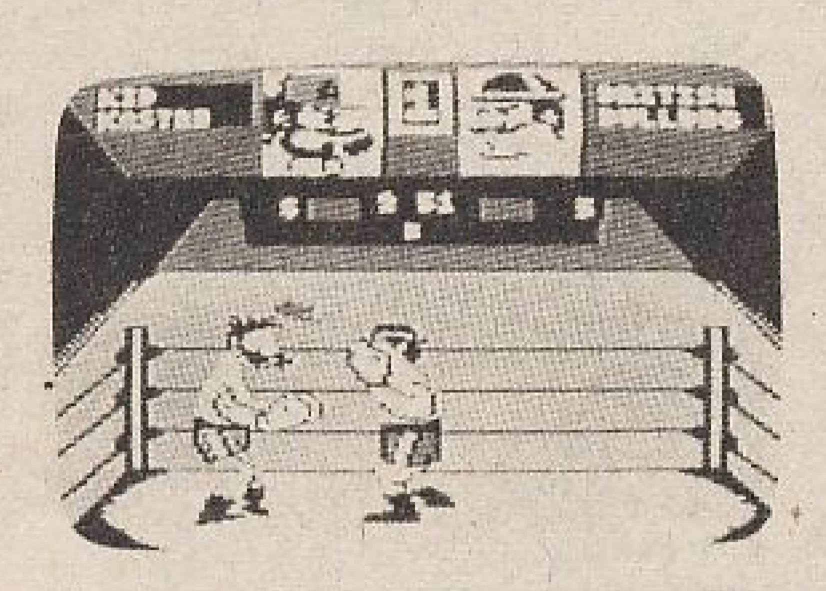
EASTERN FRONT
— 20 puntos



GYRUS
— 20 puntos



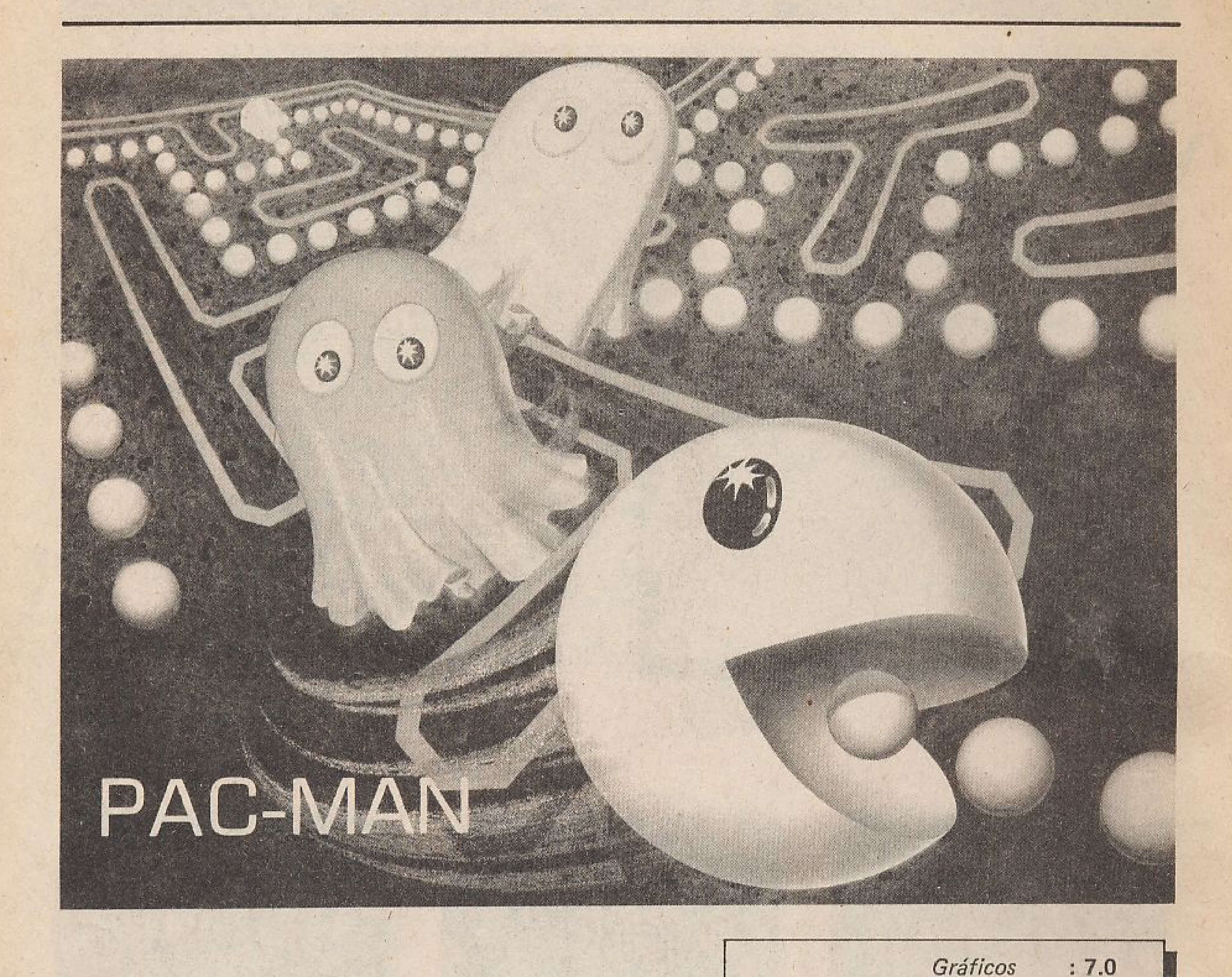
PAC-MAN
— 15 puntos



FIGHT NIGHT
— 10 puntos



WINTER GAMES (ST)
— 10 puntos



Nombre	*: PAC-MAN*
Productor	*: ATARI*
Memoria	*: 16 Kb.*

CALIFICACION (nota de 1 a 7)	*Gráficos*	**: 7.0**
	Sonido	**: 7.0**
	Animación	**: 7.0**
	Dificultad	**: 7.0**
	Originalidad	**: 7.0**
	Promedio	**: 7.0**

Este programa, sin duda alguna es un clásico de los video juegos, que tuvo gran popularidad a comienzos de nuestra década.

Para hablar de él nos remontaremos aproximadamente al año 1981, cuando apareció por primera vez en las tiendas de video en nuestro país. El impacto que produjo fue casi inmediato y al poco tiempo apareció la versión para consolas de juego.

El éxito de este juego no fue solo en nuestro país, sino que en casi todo el mundo. En Estados Unidos se, desató la llamada "Pac Man Feever', que originó diseños de juguetes, ropa, revistas e incluso una canción que hablaba de Pac Man.

El juego en si es muy simple: consiste en guiar al protagonista (Pac Man) a través de un laberinto repleto de puntos. Tu objetivo es comer todos los puntos y evitar ser capturado por los temibles Globlin. Una vez terminada esa etapa pasarás a la siguiente, que tiene mayor grado de dificultad.

Algunas tareas adicionales son:

— comer las frutas que aparecen de improviso en el centro del laberinto, las que dan puntos extras.

— contraatacar a la persecución de los fantasmas. Una vez que comes una de las 4 galletas en los vertices puedes perseguir por algunos segundos a los Globlin y ganar más puntos.

Sin duda alguna, pasarás con este juego momentos muy agradables y entretenidos como consecuencia de la fiebre de Pac Man.

Nos vemos el próximo mes.



TENGA EN CASA UN AYUDANTE DE GRAN MEMORIA Y CAPACIDAD.
ATARI 130XE, un ayudante de 8 bit que usted puede utilizar en casa para complementar su trabajo, porque el ATARI 130XE es un computador de DOBLE MEMORIA, que posee el DOBLE de la capacidad de otros computadores en su categoria, lo que se traduce en un mejor aprovechamiento de programas como:
– Planillas electrónicas
– Procesadores de texto
– Bases de datos
Además el ATARI 130XE es un excelente compañero de estudios para sus hijos, un compañero que les enseña y los entretiene con la biblioteca de programas educacionales en español más grande del mercado, basados en los programas y planes del Ministerio de Educación.
ESPECIFICACIONES TECNICAS
El ATARI 130XE es un computador de 128 Kbytes de memoria RAM, controlable por programa.
Con el Sistema Operativo en Disco DOS 2,5, el ATARI 130XE puede utilizar los 64 Kbytes adicionales de la memoria como RAMDISK.
También dispone de:
– 24 Kbytes de memoria ROM para el BASIC y es Sistema Operativo.
– Cuenta con 16 modos graficos distintos
– Un total de 256 colores
– Resolución en pantalla de 320 por 192 puntos
– 5 modos de texto diferente
– Despliega 40 caracteres por 24 líneas de texto normal
– Posee 4 sintetizadores de sonido controlables por programa
– Dispone de salida a TV o monitor de video
– Teclado profesional de 57 teclas
– 5 teclas de funciones especiales
– Letras en mayúsculas y minúsculas
– Set de caracteres en espanol y 29 caracteres gráficos
– Lenguajes de programación: BASIC, LOGO, PASCAL, FORTH, C, ASSEMBLER
ATARI-COELSA da más que los demás